

Proof of Concept Technical Solution

for the *Marconi Law Firm, LLC*.

(WordPress Website)

Project Background: Assume that you are an entrepreneur and that you own your own Information Technology (IT) consulting firm. You have recently acquired a new client called “**Marconi Law Firm**”.

As part of your client’s contract & paid agreement, you are to deliver full documentation for their upcoming WordPress Website Hosting project implementation. This documentation includes a Proof-of-Concept Technical Solution which documents all software, hardware, and network configuration details. Assume that the finished document will be used in-house by the Marconi Law Firm and will be referenced by their in-house IT department---after the project has been successfully completed.

Jalif E Vazquez Melendez

Cyber-Ware Solutions LLC.

Preface

This document will serve as proof of concept to Mr. Marconi for creating his WordPress website for his law firm and as audit documentation.

The purpose of audit documentation is to provide a comprehensive record of the organization's information technology infrastructure and security controls and processes. It plays a crucial role in providing transparency, accountability, and QA/QC regarding an organization's cybersecurity controls and practices. It enables organizations to demonstrate compliance, identify areas for improvement, and make informed decisions to strengthen their overall organizational cybersecurity.

Audit documentation serves several important purposes:

- **Compliance:** Evidence that an organization has undergone a thorough examination of its systems. It helps validate that the organization has implemented appropriate controls to protect its information systems and sensitive data.

- **Validation:** Verification of the effectiveness and adequacy of cybersecurity controls. It provides detailed information about the design, implementation, and operation of these controls, enabling reviewers to assess their reliability and identify any gaps or weaknesses.
- **Records Maintenance:** Historical record of cybersecurity audits conducted over time. It enables organizations to track their progress, identify trends, and evaluate the effectiveness actions taken. It also serves as reference for future audits and allows auditors to understand the current cybersecurity implemented and facilitates a more targeted approach to future cybersecurity updates and audits.
- **Decision-making Support:** Valuable insights and information that can support decision-making processes. It allows management to make informed decisions about allocating resources, prioritizing cybersecurity investments, and addressing identified risks and vulnerabilities.

Table of Contents

<i>Inventory</i>	7
<i>Custom Network</i>	7
<i>IDs and Passwords</i>	7
<i>Network Topology Diagram</i>	8
<i>Node.js Application (Ghost) on Docker</i>	10
Show screenshot of your CentOS 7 Console in VE	10
Update CentOS	11
Install EPEL Packages	12
Install Nano Editor	13
Docker CE	14
Install required packages	14
Install Docker CE	15
Verify docker version	16
Initialize Docker	17
Start Docker	17
Enable Docker	18
Test Docker (hello-world)	19
Disable SELinux	20
Reboot VM	21
Test SELinux	22
Confirm SELinux Status	23
Install Ghost Docker Container	24
Test Ghost	25
Ghost Container ID.....	25
<i>NginX Reverse Proxy</i>	26
Show screenshot of your Rocky 8 Console in VE	26
Update Rocky 8	27
Install EPEL Packages	28
Install Nano Editor	29
Disable SELinux	30
Reboot VM	31
Test SELinux	32

Confirm SELinux Status.....	33
Rocky Firewall.....	34
Stop Firewall	34
Disable Firewall.....	35
NginX.....	36
Install NginX	36
Start NginX	37
Enable NginX	38
Test NginX	39
Reverse Proxy for Ghost Site	40
Edit NginX configuration file	40
Reload NginX service.....	41
Terminate Docker	42
Delete Ghost Container	43
Create New Ghost Container	44
Browse to Ghost.....	45
WordPress on Ubuntu - LAMP Stack.....	46
Show screenshot of your Ubuntu Console in VE	46
Update Ubuntu	47
Upgrade Ubuntu	48
Install Nano Editor	49
Install Git	50
Install Apache2	51
Open Firewall Ports 80 and 443.....	52
Browse to Apache2 Ubuntu Default Page.....	53
Install MySQL.....	54
Alter root user password	55
Flush Privileges.....	55
Quit MySQL.....	57
Install PHP.....	57
Edit Sources.list File	58
Update Ubuntu (refreshes the repolist).....	59
Install Required PHP Libraries.....	61
Install Required MySQL Libraries	62
Enable URL Rewrites (clean URLs).....	63
Restart Apache Service	64
Create a test.php Web Page	65
Test the test.php Web Page.....	66
Database Configuration in MySQL Log into MySQL Database	67
Create WordPress Database in MySQL	67
Create WordPress User for MySQL Database	69

Grant Privileges to this New WordPress User	70
Flush Privileges.....	71
Quit MySQL	72
Install WordPress	73
Grant Permission to html Directory to WordPress User	73
Delete Files from html Directory.....	74
Verify html Directory is Empty	75
Clone WordPress to html Directory	76
Verify html Directory Contains WordPress Files	77
Verify Permissions on html Directory	78
Edit Ownership	78
Edit Ownership of Contents of html Directory.....	79
Edit Ownership of the html Directory Itself	79
Edit the apache2.conf File	81
Override All Default Apache Directives.....	83
Create a .htaccess File in the /var/www/html/.git/ Directory	84
Restart the Apache Service.....	85
WordPress Configuration	86
Configure WordPress	86
WordPress Configuration Selections.....	86
Run Installation	88
Create an Admin WordPress User	88
WordPress Site Selections.....	90
Test WordPress	91
WordPress Security Settings and Configurations	92
WordPress Security Summary	Error! Bookmark not defined.
Defense-in-depth	Error! Bookmark not defined.
File Permissions	Error! Bookmark not defined.
Vulnerability.....	92
Configuration	92
Validation.....	92
Securing wp-config.php.....	Error! Bookmark not defined.
Vulnerability.....	Error! Bookmark not defined.
Configuration	Error! Bookmark not defined.
Validation.....	Error! Bookmark not defined.
Firewall (Shield)	93
Vulnerability.....	Error! Bookmark not defined.
Configuration	Error! Bookmark not defined.
Validation.....	Error! Bookmark not defined.
Conclusion	Error! Bookmark not defined.

Appendix A	94
NginX Config File	94
Appendix B	95
NginX Access Log File	95
NginX Error Log File	96

Inventory

EQUIPMENT	OPERATING SYSTEM	ADDITIONAL INFO	IP ADDRESS
Router/Custom Network	-	-	10.10.229.1
Docker	CentOS 7	Ghost Container	10.10.229.11
NginX Reverse Proxy	Rocky 8	Reverse Proxy	10.10.229.10
WordPress	Ubuntu	LAMP Stack running WordPress	10.10.229.12

Custom Network

NETWORK NAME	SUBNET IP	SUBNET MASK	DNS	GATEWAY
ITE229	10.10.229.0	255.255.255.0	10.10.229.1	10.10.229.1

IDs and Passwords

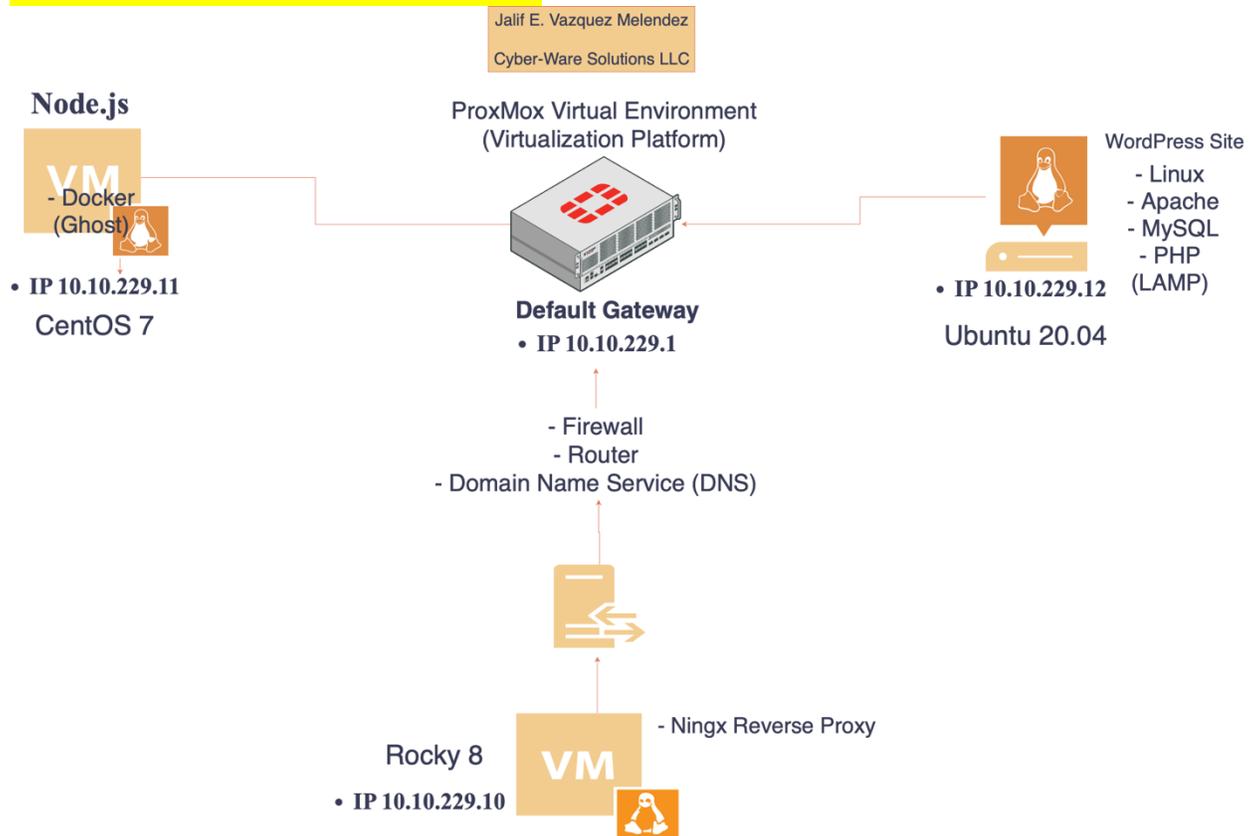
ACCOUNT	USER ID	PASSWORD
CentOS 7 Root User	root	Fullsail1!
Rocky 8 Root User	root	Fullsail1!
MySQL Root User	root@localhost	[randompassword]
MySQL WordPress User	WordPressUser	[randompassword]
WordPress Admin	admin	[randompassword]

IMPORTANT: To earn full credit, original formatting of this document must remain in place. In addition, ALL screenshots must include a “full view”, including your ProxMox title bar with your username showing.

Network Topology Diagram

IMPORTANT: Be sure your name is at the TOP of your diagram. Diagrams without your name will result in an automatic zero (0) grade for *each weekly milestone* (Weeks 1-4).

1. **CREATE & INSERT YOUR OWN IMAGE OF A “STAR” TOPOLOGY HERE – USE DRAW.IO, VISIO, WHATEVER DIAGRAMING TOOL YOU PREFER.**



2. Based on what you’ve learned and the research you’ve conducted during Week #1, write at least a **75-word summary** (or more words) explaining the purpose of each Virtual Machine (VM) to Mr. Marconi: 1) how the network traffic works, 2) software is installed on each machine, and 3) how the firewall works. Remember, that Mr. Marconi is a Lawyer and not necessarily technical. This means that your summary should be easy to understand for a non-technical Client.

Our team has the idea of bringing your desired deliverables for this project to fruition in a robust yet understandable and manageable. The ProxMox Virtual Environment is a software solution that allows you to create and manage virtual machines (VMs) and containers. It provides a platform for running multiple operating systems that is the same as computers and applications in a single physical place. Simply put, it is a virtualization platform that helps you optimize your computer hardware resources by consolidating various of them on a single place we call a server. We are using that environment to install the necessary systems to provide the project expectations while ensuring we provide performance, security, and scalability.

I emphasize the importance of what we are installing over the next four weeks and installing Node.js. Let's begin by explaining what Node.js is and the scalable network infrastructure we will use to run our applications. The second application would be Docker, and among other applications, we are installing Ghost and Nginx. Still, we are installing Nginx as a reverse proxy in our upcoming project, as it can be used in many ways. Nginx as a reverse proxy is essential because it helps to improve performance, scalability, and security. By handling incoming requests and directing them to the appropriate backend server, Nginx ensures that our application is secure and optimized for maximum performance. By installing Node.js, we can ensure that our project runs smoothly and efficiently. By doing so, we can benefit from several advantages that will help us build a high-performance and secure application. In summary, by installing Docker as a Node.js to use it with Ghost and Nginx as a reverse proxy, we can build a high-performance, secure, and scalable application that meets the law firm's needs. Ghost is a modern publishing platform that allows us to create and manage content.

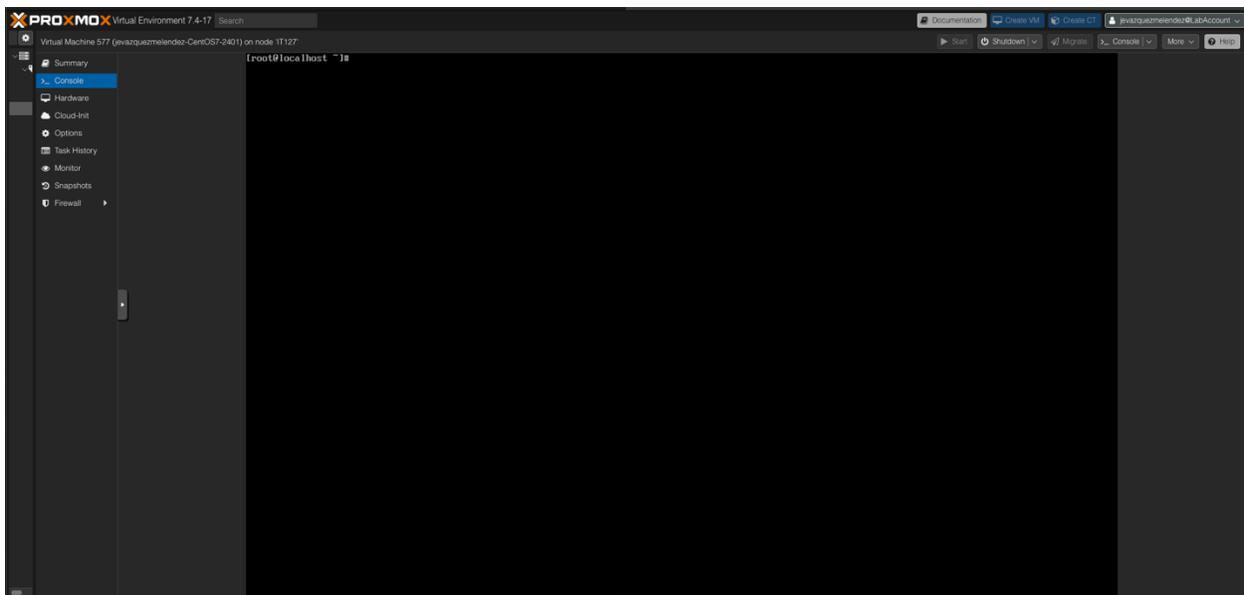
Your diagram & summary should include the following details:

- Include all VMs, IP Addresses, Software installation on each machine, Firewall, and arrows/lines to show the network traffic flow & other connections based on FSO requirements;
 - Once you have created your own Network Topology Diagram and Summary, DELETE the notes above and replace it with your own quick “network traffic summary” for the Marconi Law Firm.
 - Plan to update Topology Summary and Diagram during each weekly milestone---as we ADD software installations & server configurations to this project.
 - **NOTE:** If you need additional info “**STAR**” Network Topology Diagrams, please do some research online---starting with links below:
<https://www.linkedin.com/learning/cisco-networking-foundations-fundamentals-of-cisco-networking/examples-of-network-topologies?autoplay=true&resume=false&u=50813145>
<https://www.comparitech.com/net-admin/network-topologies-advantages-disadvantages/>
-

IMPORTANT: To earn full credit, original formatting of this document must remain in place. In addition, ALL screenshots must include a “full view”, including your ProxMox title bar with your username showing.

Milestone #1: Node.js App (Ghost) on Docker

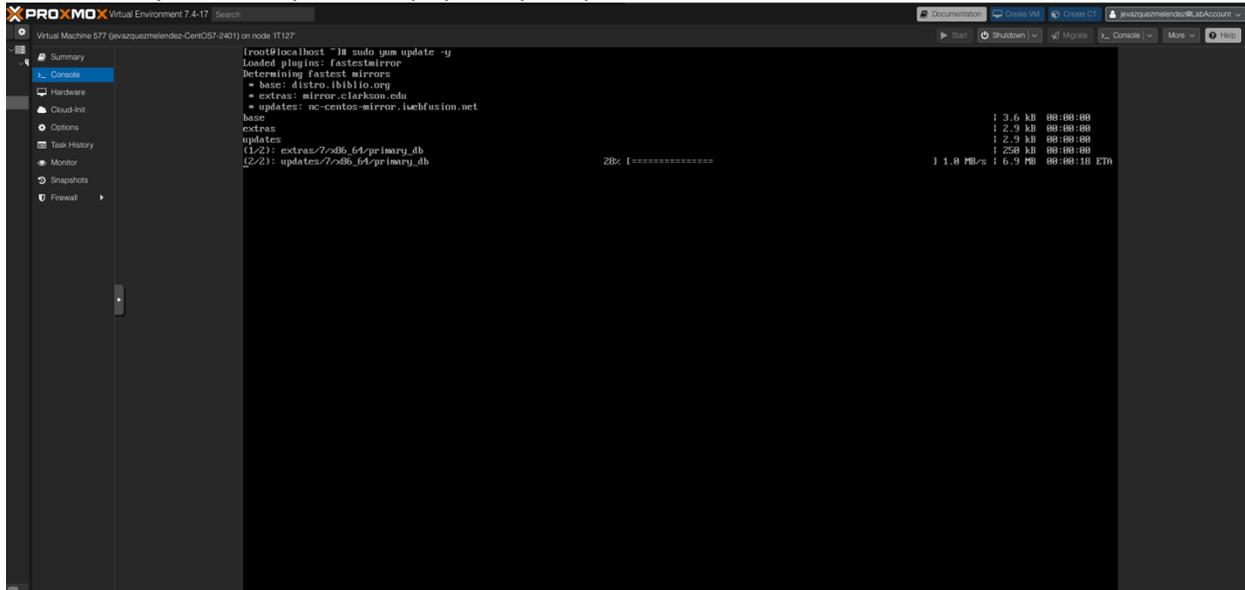
Show screenshot of your CentOS 7 Console in VE



Update CentOS

```
# sudo yum update -y
```

The command `sudo yum update -y` is used to update the installed packages on your system. The `-y` flag automatically answers "yes" to any update prompts.

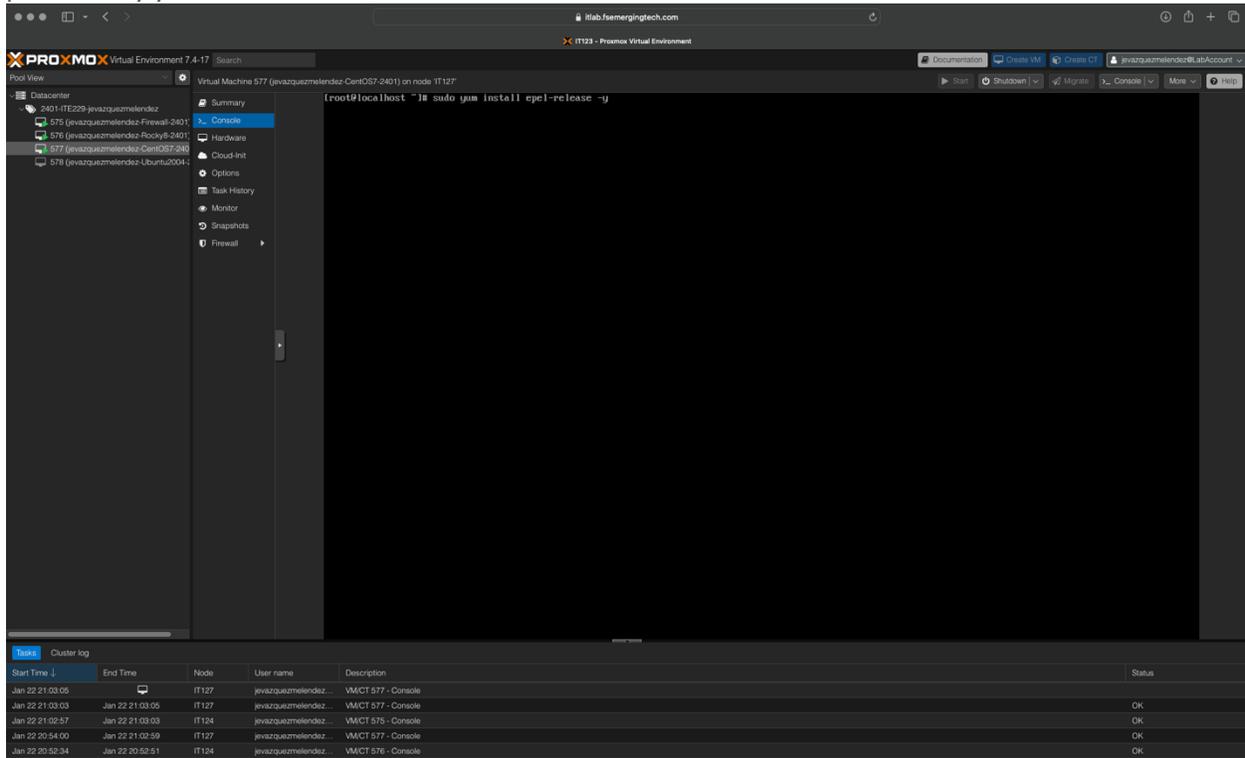


```
[root@localhost ~]# sudo yum update -y
Loaded plugins: fastestmirror
Determining fastest mirrors
 * base: distro.biblio.org
 * extras: mirror.clarkson.edu
 * updates: nc-centos-mirror.luchfusion.net
base                                     1.3.6 kB  00:00:00
extras                                  1.2.9 kB  00:00:00
updates: (1/2): extras/7/x86_64/primary.db 1.2.9 kB  00:00:00
updates: (2/2): updates/7/x86_64/primary.db 1.250 kB  00:00:00
ZB: (=====) 1 1.0 MB/s 1 6.9 MB 00:00:10 ETA
```

Install EPEL Packages

```
# sudo yum install epel-release -y
```

The command `sudo yum install epel-release` installs your system's Extra Packages for Enterprise Linux (EPEL) repository. This repository contains additional software packages not included in the default repositories provided by your Linux distribution.



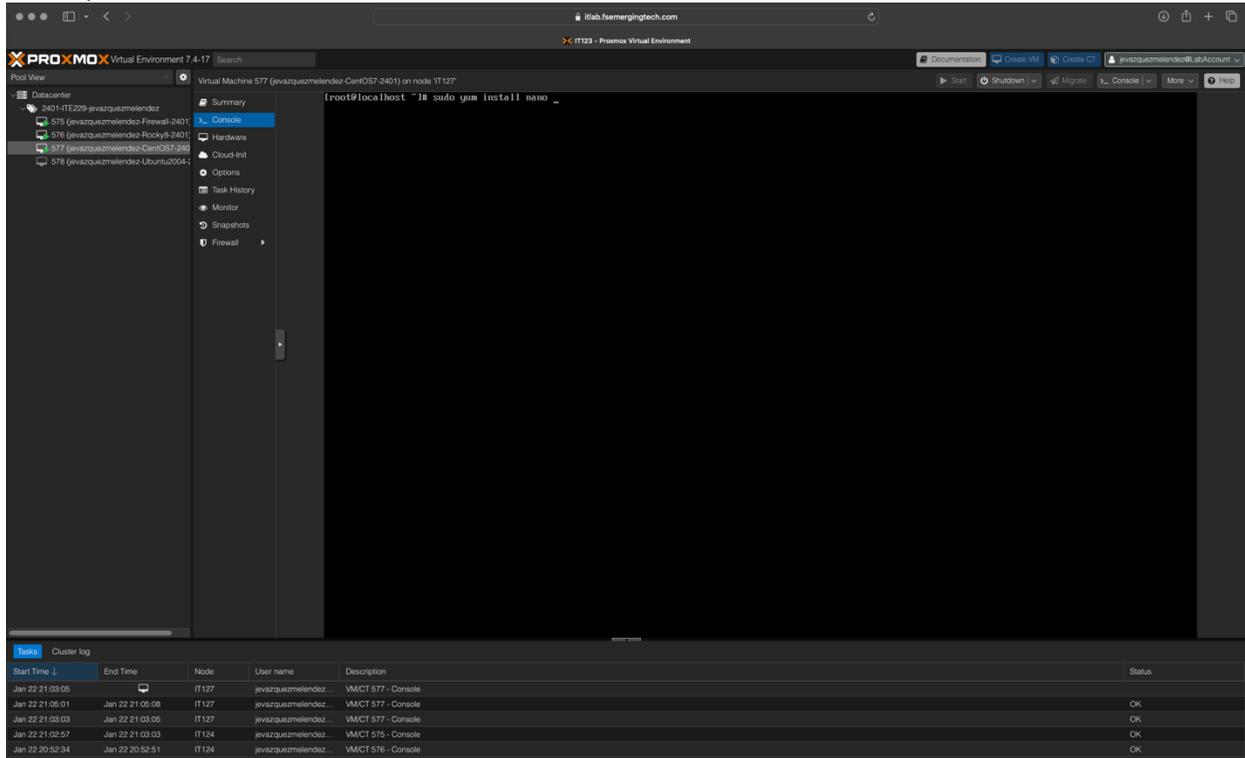
The screenshot displays the Proxmox Virtual Environment (VE) interface. The main window shows a terminal session for VMCT 577 on node IT127, where the command `sudo yum install epel-release -y` has been executed. The interface includes a sidebar with navigation options like Summary, Console, Hardware, Cloud-Init, Options, Task History, Monitor, Snapshots, and Firewall. At the bottom, a 'Tasks' table provides a log of recent operations.

Start Time	End Time	Node	User name	Description	Status
Jan 22 21:03:05		IT127	jevaquizmelendez...	VMCT 577 - Console	
Jan 22 21:03:03	Jan 22 21:03:05	IT127	jevaquizmelendez...	VMCT 577 - Console	OK
Jan 22 21:02:57	Jan 22 21:03:03	IT124	jevaquizmelendez...	VMCT 575 - Console	OK
Jan 22 20:54:03	Jan 22 21:02:59	IT127	jevaquizmelendez...	VMCT 577 - Console	OK
Jan 22 20:52:34	Jan 22 20:52:51	IT124	jevaquizmelendez...	VMCT 576 - Console	OK

Install Nano Editor

```
$ sudo yum install nano -y
```

The command `sudo yum install nano` installs the Nano text editor on your system. Nano is a simple, user-friendly text editor that can create and edit text files.



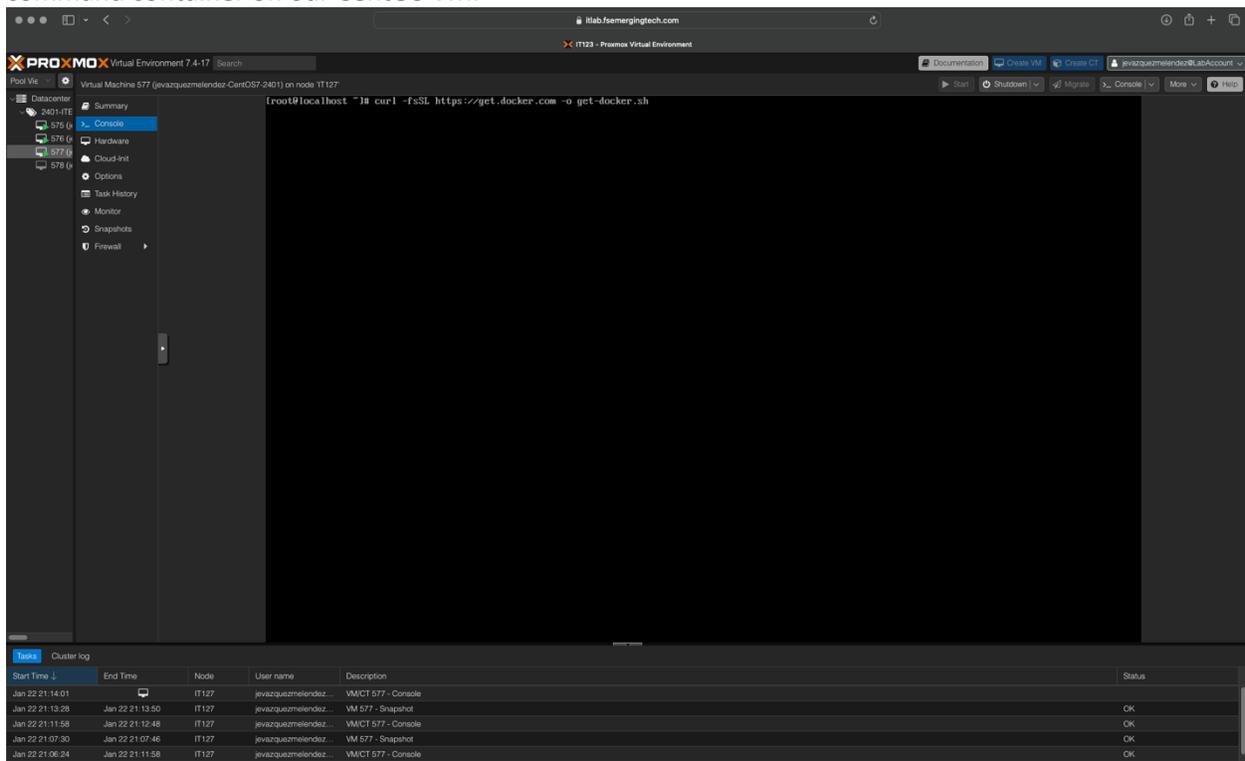
Docker CE

Install required packages

Type the following commands to download docker ce.

```
# sudo -fsSL https://get.docker.com -o get-docker.sh
```

These commands, typed as follows, allow us to install all the necessary packages to use the docker This command container on our CentOS VM.



The screenshot shows the Proxmox VE interface. The main window displays a terminal for a virtual machine named 'jevizquzmelendez-CentOS7-2401' on node 'IT127'. The terminal prompt is '[root@localhost ~]# curl -fsSL https://get.docker.com -o get-docker.sh'. The terminal output is mostly black, indicating the command is running. The interface also shows a task log at the bottom.

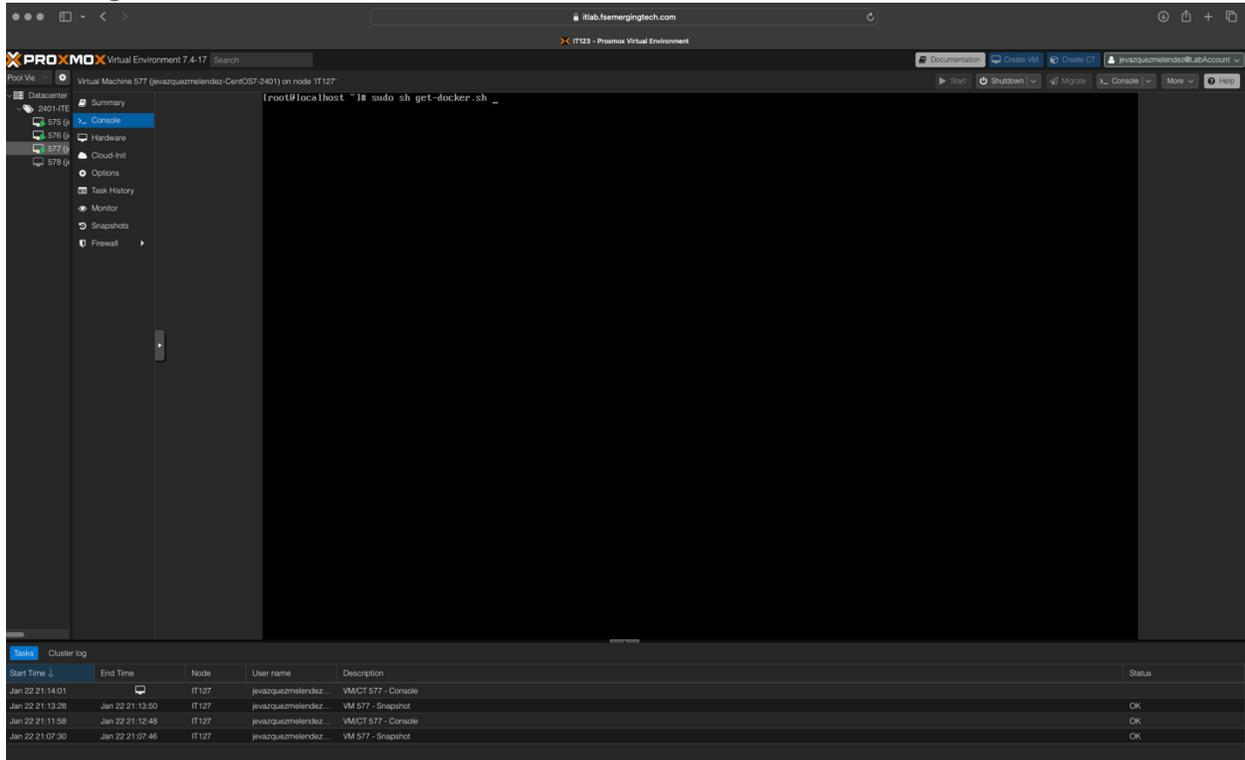
Start Time	End Time	Node	User Name	Description	Status
Jan 22 21:14:01		IT127	jevizquzmelendez...	VMCT 577 - Console	
Jan 22 21:13:28	Jan 22 21:13:50	IT127	jevizquzmelendez...	VM 577 - Snapshot	OK
Jan 22 21:11:58	Jan 22 21:12:48	IT127	jevizquzmelendez...	VMCT 577 - Console	OK
Jan 22 21:07:30	Jan 22 21:07:46	IT127	jevizquzmelendez...	VM 577 - Snapshot	OK
Jan 22 21:06:24	Jan 22 21:11:58	IT127	jevizquzmelendez...	VMCT 577 - Console	OK

Install Docker CE

To install the latest version, run the following:

```
# sudo sh get-docker.sh
```

This will grab the latest version of Docker and install it.

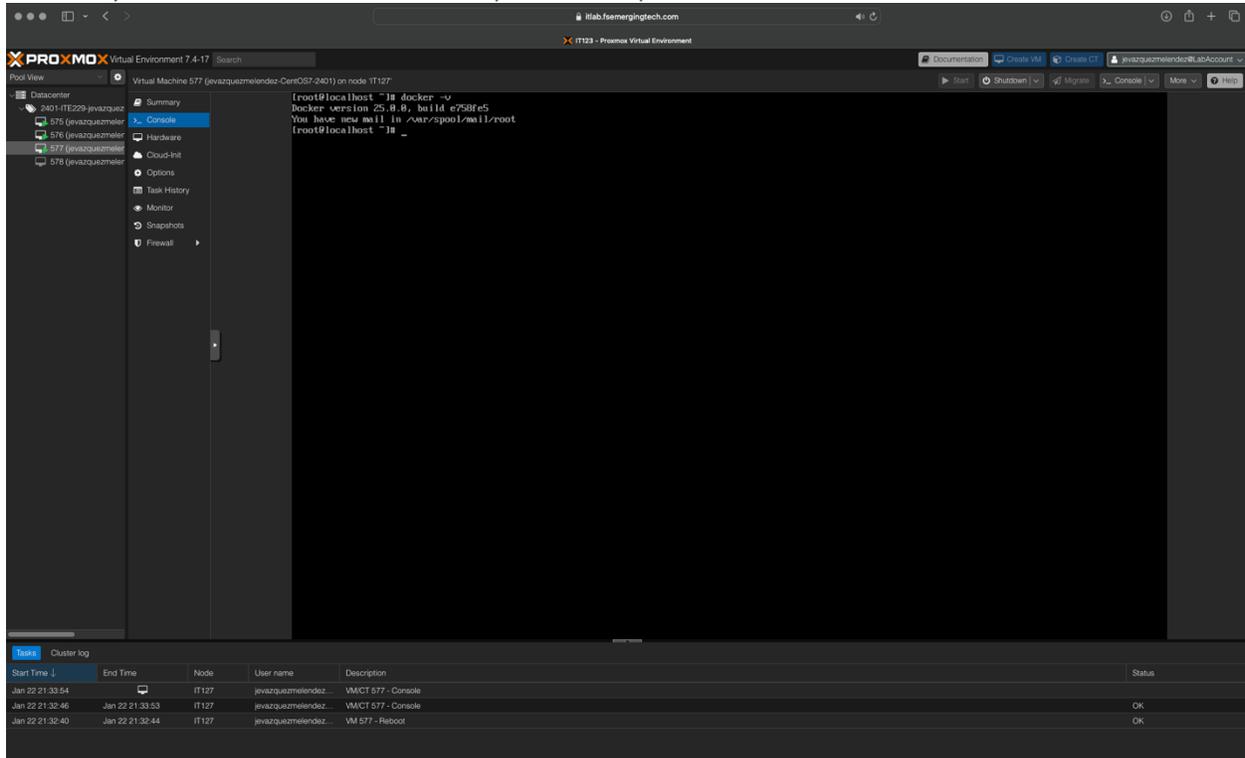


The screenshot shows the Proxmox VE interface. The main window displays a terminal session for Virtual Machine 577 on node IT127. The terminal prompt is `[root@localhost ~]#` and the command `sudo sh get-docker.sh` has been entered. The left sidebar shows the navigation menu with 'Console' selected. The bottom of the interface displays a 'Tasks' table with the following data:

Start Time	End Time	Node	User name	Description	Status
Jan 22 21:14:01		IT127	jevizquezmelendez	VMCT 577 - Console	
Jan 22 21:13:28	Jan 22 21:13:50	IT127	jevizquezmelendez	VM 577 - Snapshot	OK
Jan 22 21:11:58	Jan 22 21:12:48	IT127	jevizquezmelendez	VMCT 577 - Console	OK
Jan 22 21:07:30	Jan 22 21:07:46	IT127	jevizquezmelendez	VM 577 - Snapshot	OK

Verify docker version
docker -v

This command is used to verify the version of the current docker program installed. We can stay up to date and verify the docker version installed in previous steps with this command.



The screenshot shows a Proxmox VE interface with a terminal window open. The terminal displays the output of the 'docker -v' command, which is 'Docker version 25.0.0, build e798fe5'. The terminal prompt is '[root@localhost ~]#'. The Proxmox interface includes a sidebar with navigation options like Summary, Console, Hardware, Cloud-Init, Options, Task History, Monitor, Snapshots, and Firewall. At the bottom, there is a 'Tasks' table showing a list of recent operations.

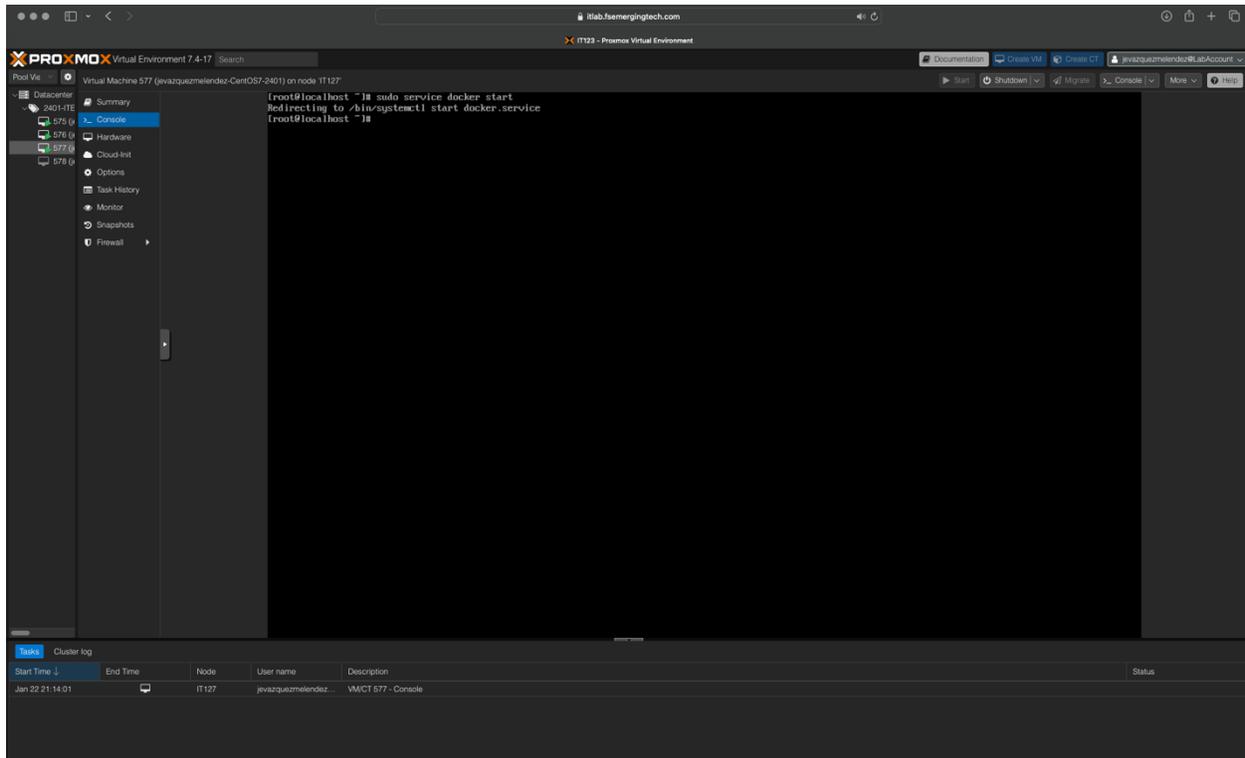
Start Time	End Time	Node	User Name	Description	Status
Jan 22 21:33:54		ft127	jevaquzmelendez...	VM/CT 577 - Console	
Jan 22 21:32:46	Jan 22 21:33:53	ft127	jevaquzmelendez...	VM/CT 577 - Console	OK
Jan 22 21:32:40	Jan 22 21:32:44	ft127	jevaquzmelendez...	VM 577 - Reboot	OK

Initialize Docker

Start Docker

```
# sudo systemctl start docker
```

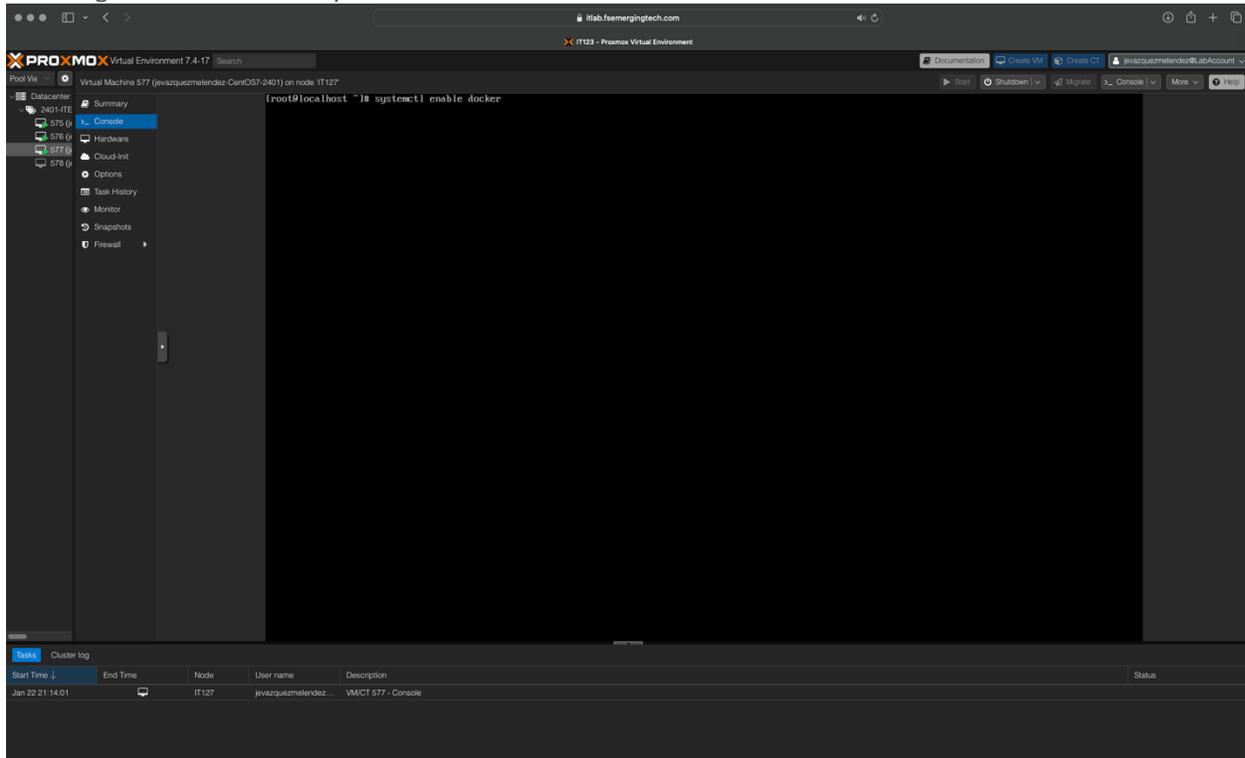
The command `sudo systemctl start docker` starts the Docker service on a Linux system, allowing you to run Docker containers.



Enable Docker

```
# sudo systemctl enable docker
```

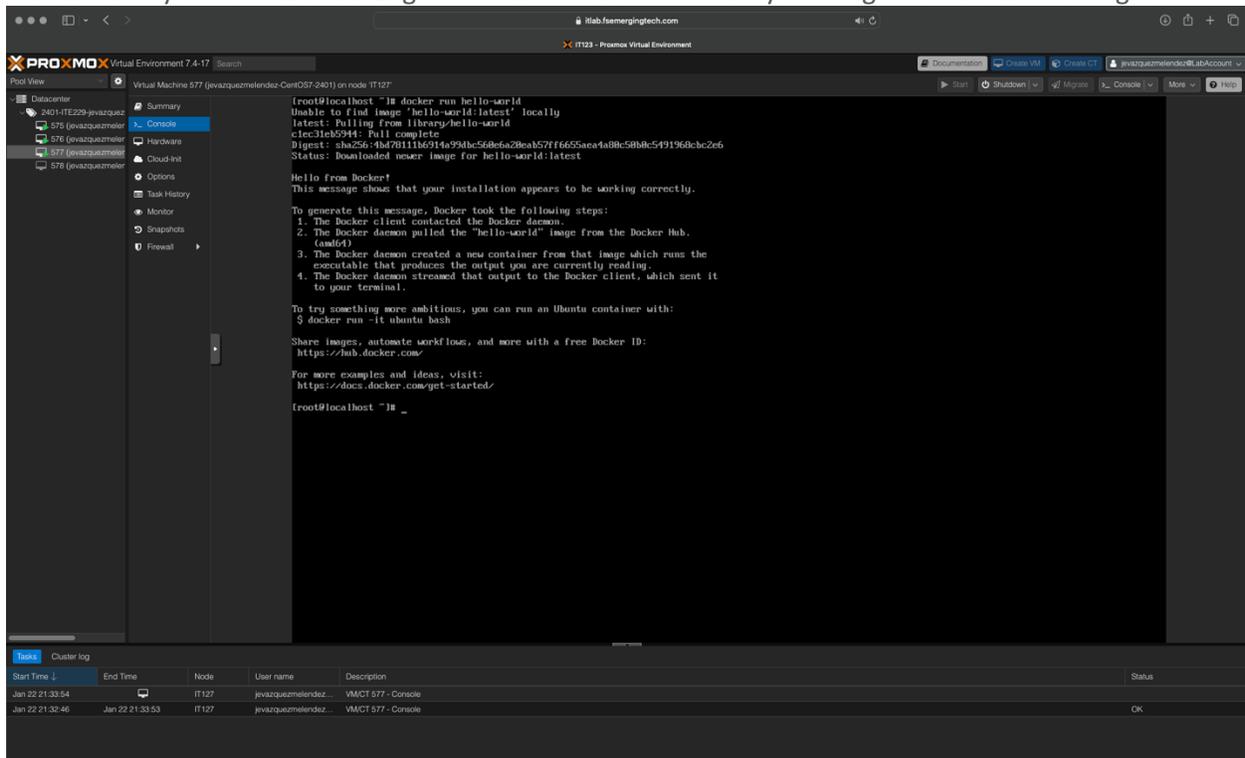
The command `sudo systemctl` enables us to set docker to start automatically when the system boots up, ensuring the service is always available.



Test Docker (hello-world)

```
# sudo docker run hello-world
```

This will verify that the Docker Engine installation is successful by running the hello-world image.



The screenshot shows a terminal window within a Proxmox Virtual Environment. The terminal output is as follows:

```
root@localhost:~# docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
c1ec31b52914: Pull complete
Digest: sha256:4bd781116914a994bc5686a29ea357ff6655a6a4a88c588b0c5491968cb2e6
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (cached).
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

root@localhost:~# _
```

At the bottom of the terminal window, there is a 'Tasks' table with the following data:

Start Time	End Time	Node	User name	Description	Status
Jan 22 21:33:54		IT127	javaquzmelendez...	VMCT 577 - Console	
Jan 22 21:32:46	Jan 22 21:33:53	IT127	javaquzmelendez...	VMCT 577 - Console	OK

Disable SELinux

```
# sudo nano /etc/selinux/config
```

To disable SELinux by editing its configuration file, you can follow these steps:

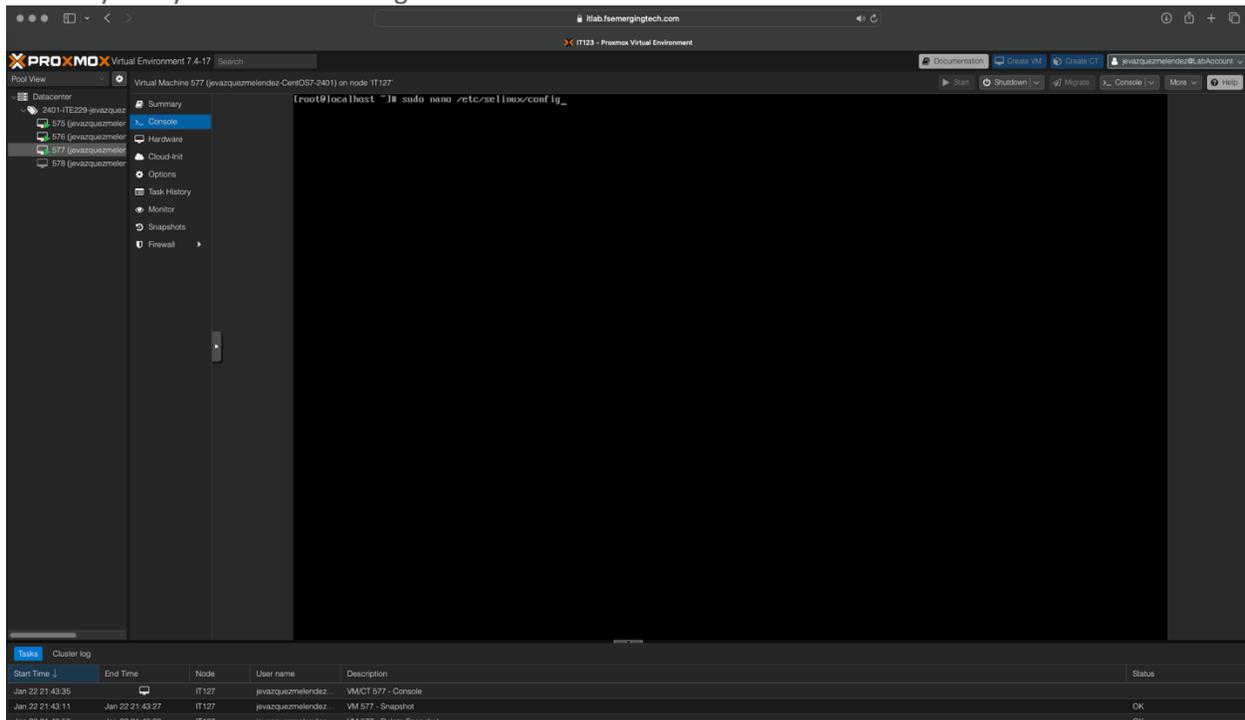
Run the following command to open the SELinux configuration file in the Nano text editor with root privileges:

```
# sudo nano /etc/selinux/config
```

Locate the line that starts with SELINUX= and change its value to disabled.

Save the changes by pressing Ctrl + X and pressing enter.

Reboot your system for the changes to take effect.

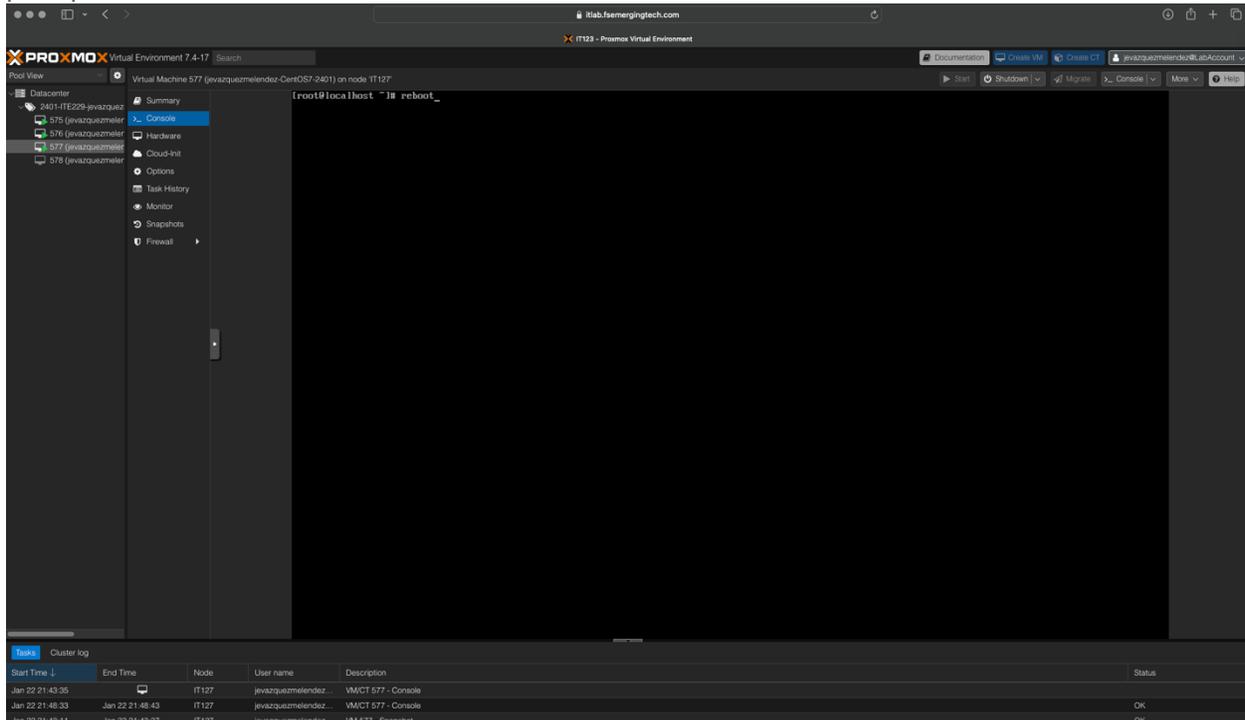


Reboot VM

Reboot VM

```
# reboot
```

This command is used to reboot or reset a Virtual Machine from the terminal shell using a command prompt.

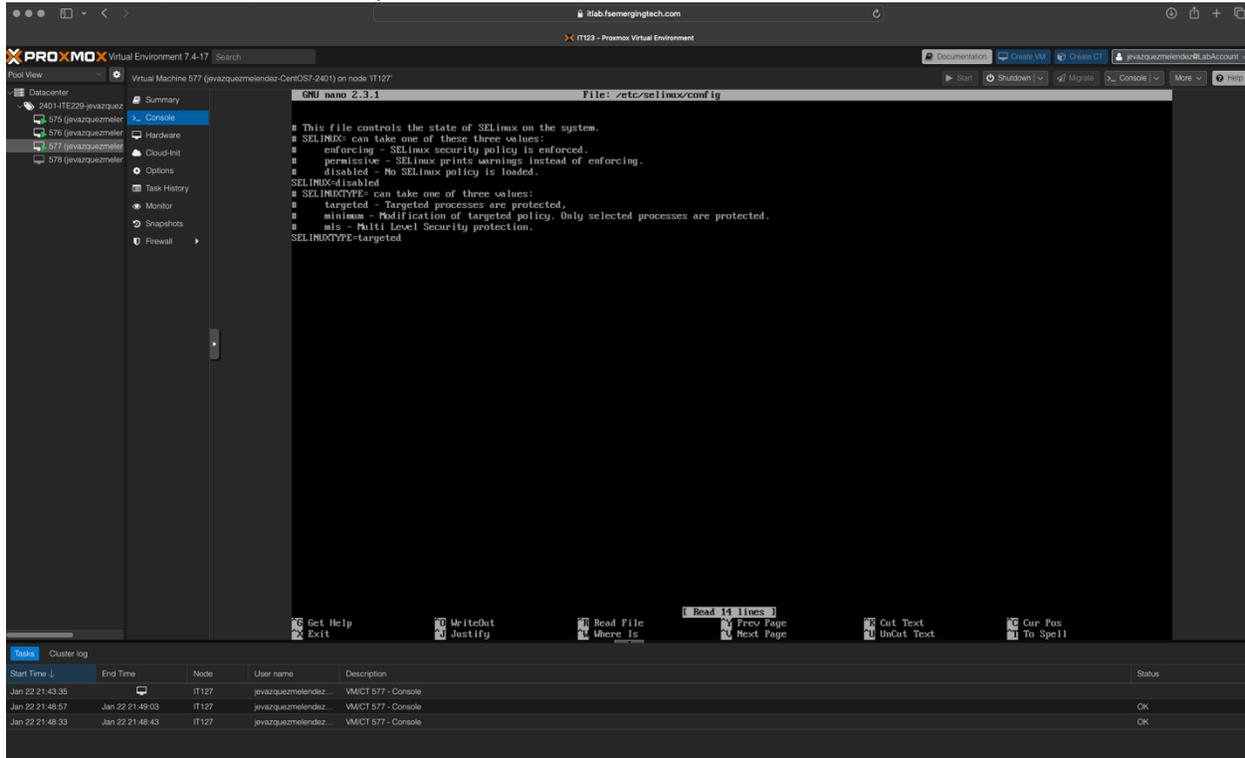


Test SELinux

```
# sudo nano /etc/selinux/config
```

Locate the line that starts with SELINUX= and test if it remains disabled.

Press Ctrl + X, then Y, and finally Enter.



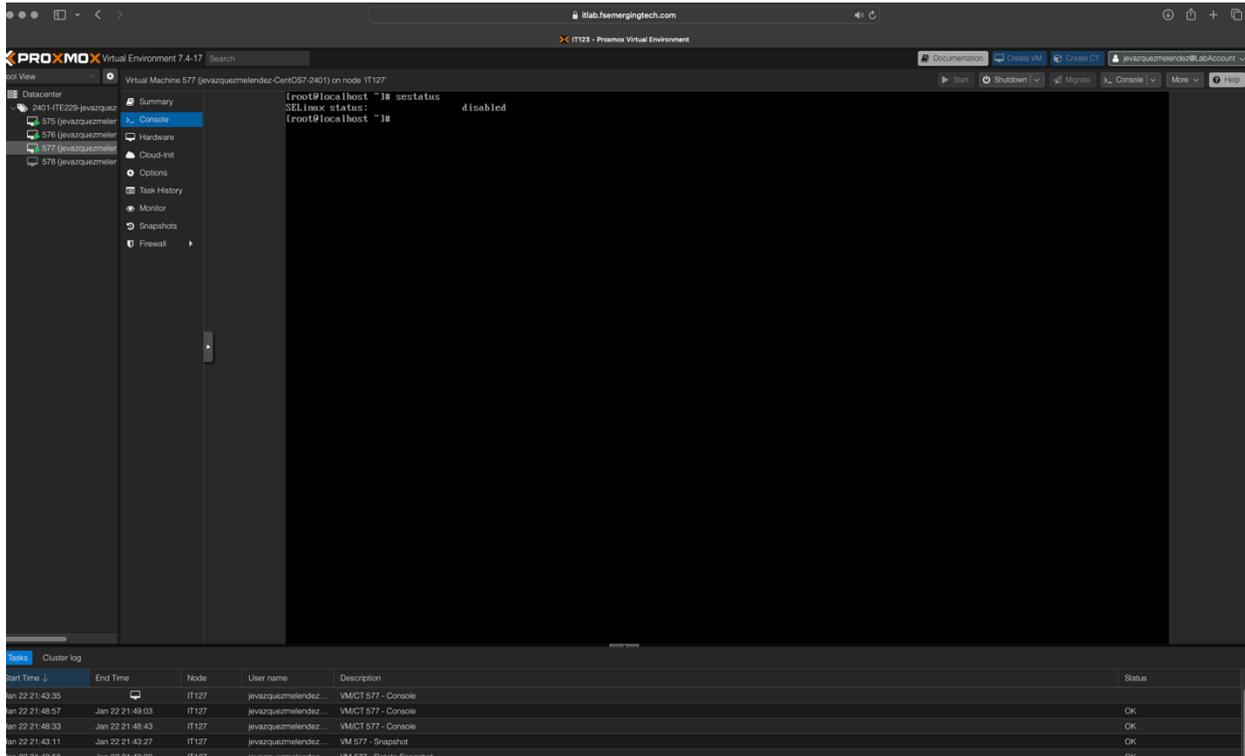
```
GNU nano 2.3.1 File: /etc/selinux/config
# This file controls the state of SELinux on the system.
# SELINUX can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - No SELinux policy is loaded.
SELINUX=disabled
# SELINUXTYPE can take one of three values:
#   targeted - Targeted processes are protected.
#   minimum - Modification of targeted policy. Only selected processes are protected.
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

Start Time	End Time	Node	User name	Description	Status
Jan 22 21:43:35		IT127	jevaquazmelendez...	VMCT 577 - Console	
Jan 22 21:48:57	Jan 22 21:49:03	IT127	jevaquazmelendez...	VMCT 577 - Console	OK
Jan 22 21:48:33	Jan 22 21:48:43	IT127	jevaquazmelendez...	VMCT 577 - Console	OK

Confirm SELinux Status

sestatus

This command is used to check the status of the SELinux (Security-Enhanced Linux) system on a Linux machine.



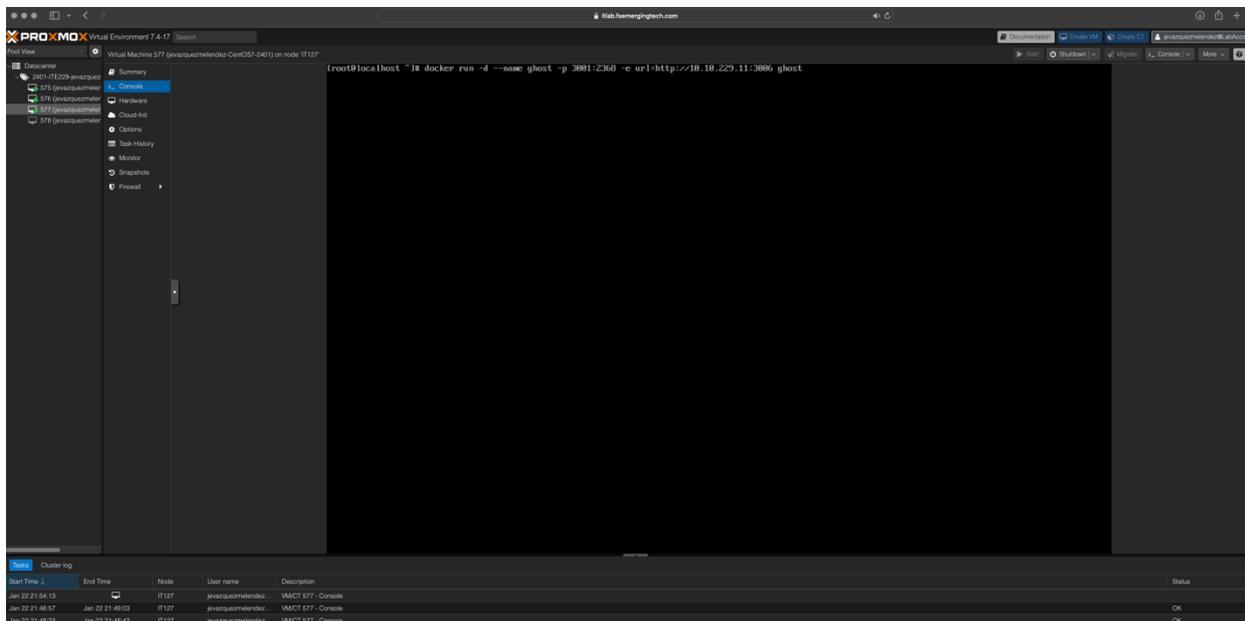
The screenshot shows a Proxmox Virtual Environment (VE) interface. The main window displays a terminal session for a virtual machine named '577 (jevazquezmelendez-CentOS-7401)' on node 'IT127'. The terminal output shows the command '# sestatus' being executed, resulting in the output 'SELinux status: disabled'. The terminal prompt is '[root@localhost ~]#'. The interface includes a sidebar with navigation options like Summary, Console, Hardware, Cloud-Init, Options, Task History, Monitor, Snapshots, and Firewall. At the bottom, there is a 'Tasks' table showing a cluster log.

Start Time ↓	End Time	Node	User name	Description	Status
Jan 22 21:43:35		IT127	jevazquezmelendez	VMCT 577 - Console	
Jan 22 21:48:57	Jan 22 21:49:03	IT127	jevazquezmelendez	VMCT 577 - Console	OK
Jan 22 21:48:33	Jan 22 21:48:43	IT127	jevazquezmelendez	VMCT 577 - Console	OK
Jan 22 21:43:11	Jan 22 21:43:27	IT127	jevazquezmelendez	VM 577 - Snapshot	OK
Jan 22 21:42:58	Jan 22 21:43:00	IT127	jevazquezmelendez	VM 577 - Delete Snapshot	OK

Install Ghost Docker Container

```
# docker run -d --name ghost -p 3001/2368 -e url=http://10.10.229.11/blog ghost
```

This command runs a containerized application called "ghost," a blogging platform. The `-d` flag starts the container in detached mode, meaning it runs in the background. The `--name` flag is used to give the container a name, in this case, "ghost." The `-p` flag maps the host machine's port `3001` to the container's port `2368`. This is done so the blog can be accessed from the host machine's browser by going to `http://localhost:3001/blog` domain. The `-e` flag sets an environment variable called URL, which specifies the blog's URL. In this case, it is set to `http://10.10.229.11/blog`. Finally, "ghost" at the end of the command specifies the name of the Docker image for the container.

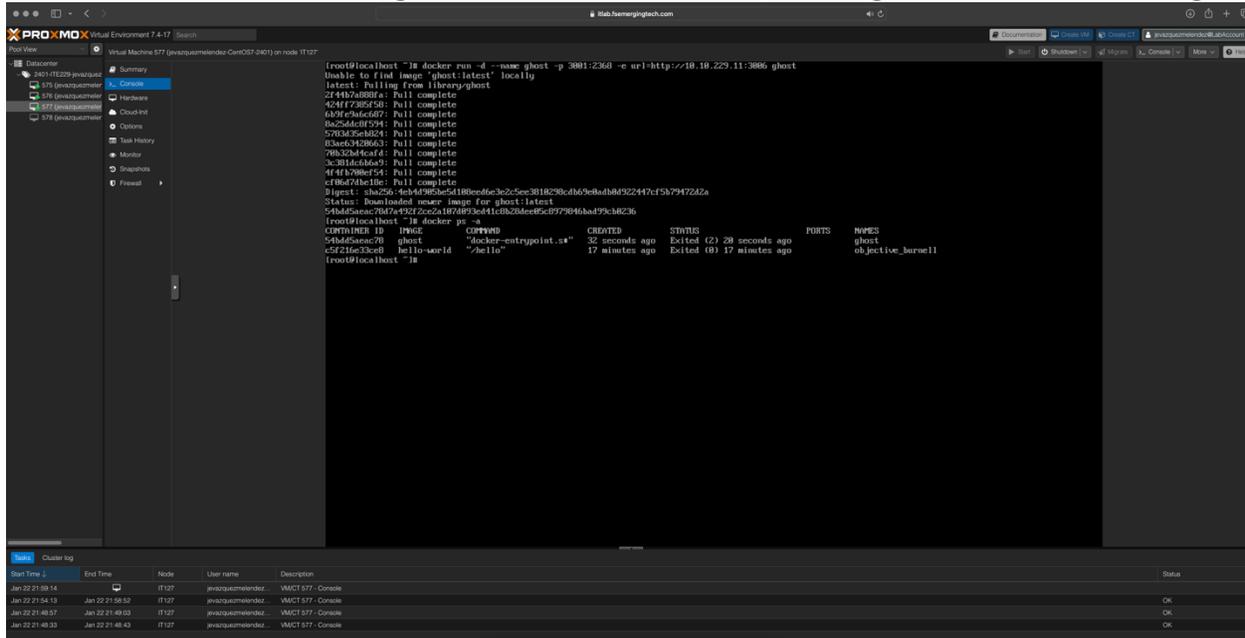


Test Ghost

Ghost Container ID

docker ps -a

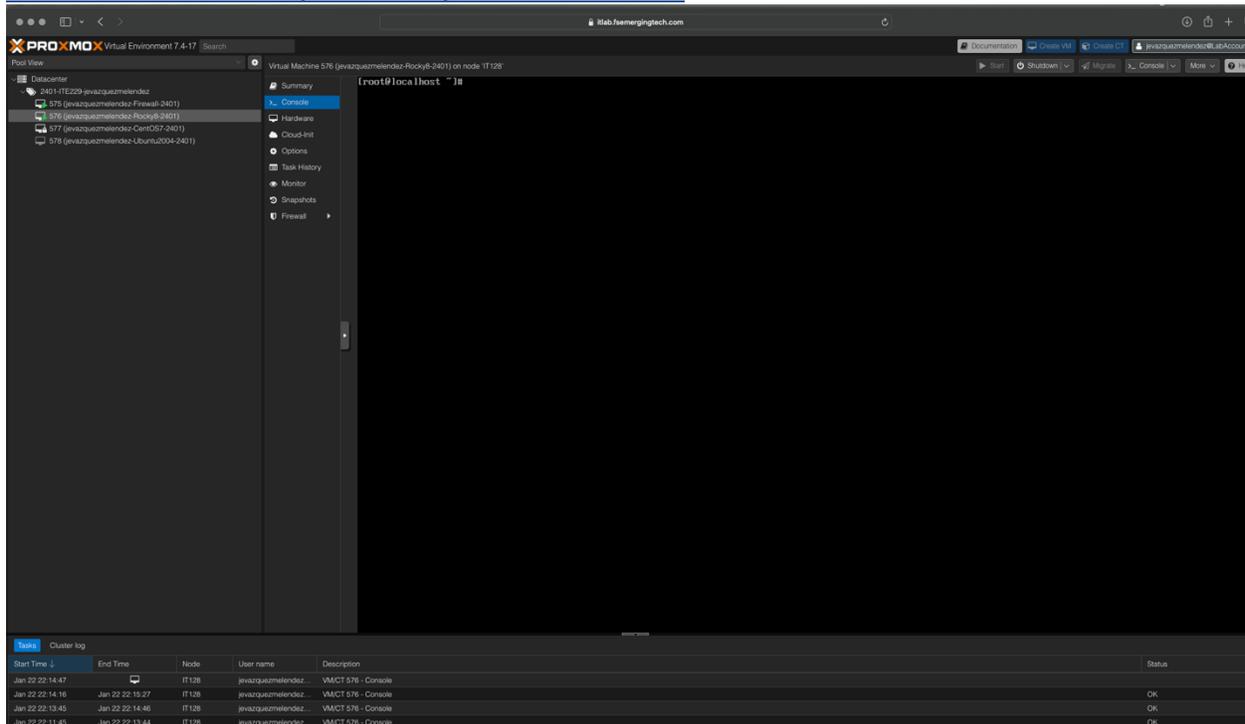
The command `docker ps -a` lists all the containers that have been created on the system, including their status, container ID, image used to create the container, command that was used, creation time, and name of the container. The `-a` flag is used to show all containers, including those that are not running.



IMPORTANT: To earn full credit, original formatting of this document must remain in place. In addition, ALL screenshots must include a “full view”, including your ProxMox title bar with your username showing.

Milestone #1: NginX Reverse Proxy

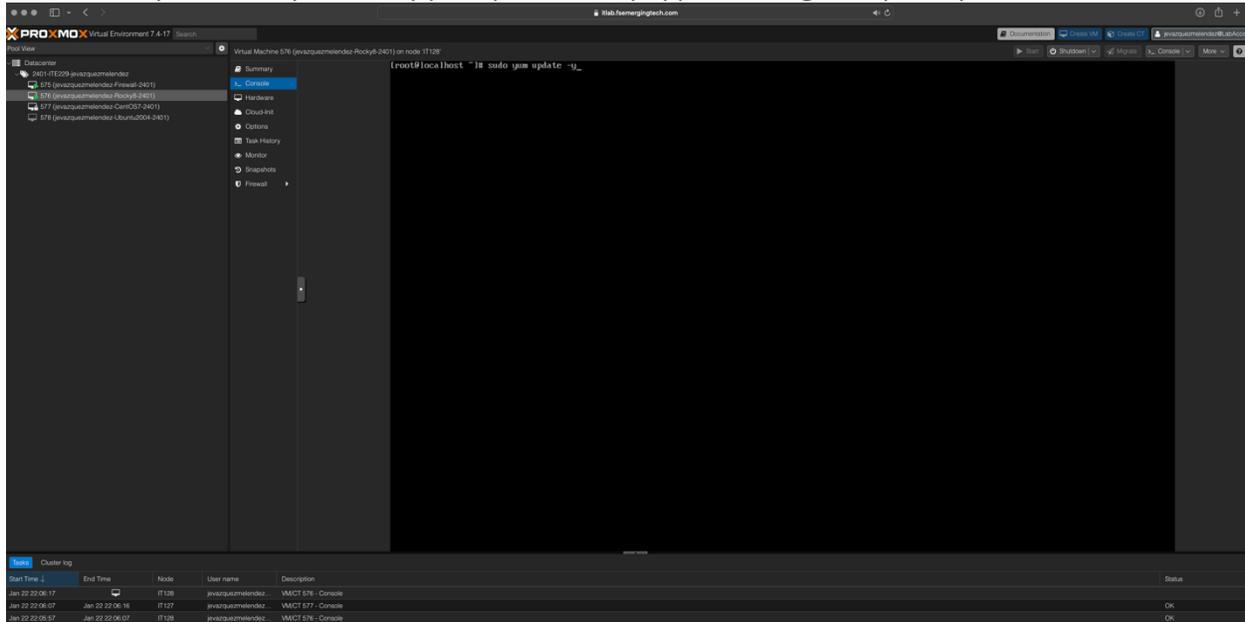
Show screenshot of your Rocky 8 Console in VE



Update Rocky 8

```
# sudo yum update -y
```

The command `sudo yum update -y` is used to update the installed packages on your system. The `-y` flag automatically answers "yes" to any prompts that may appear during the update process.



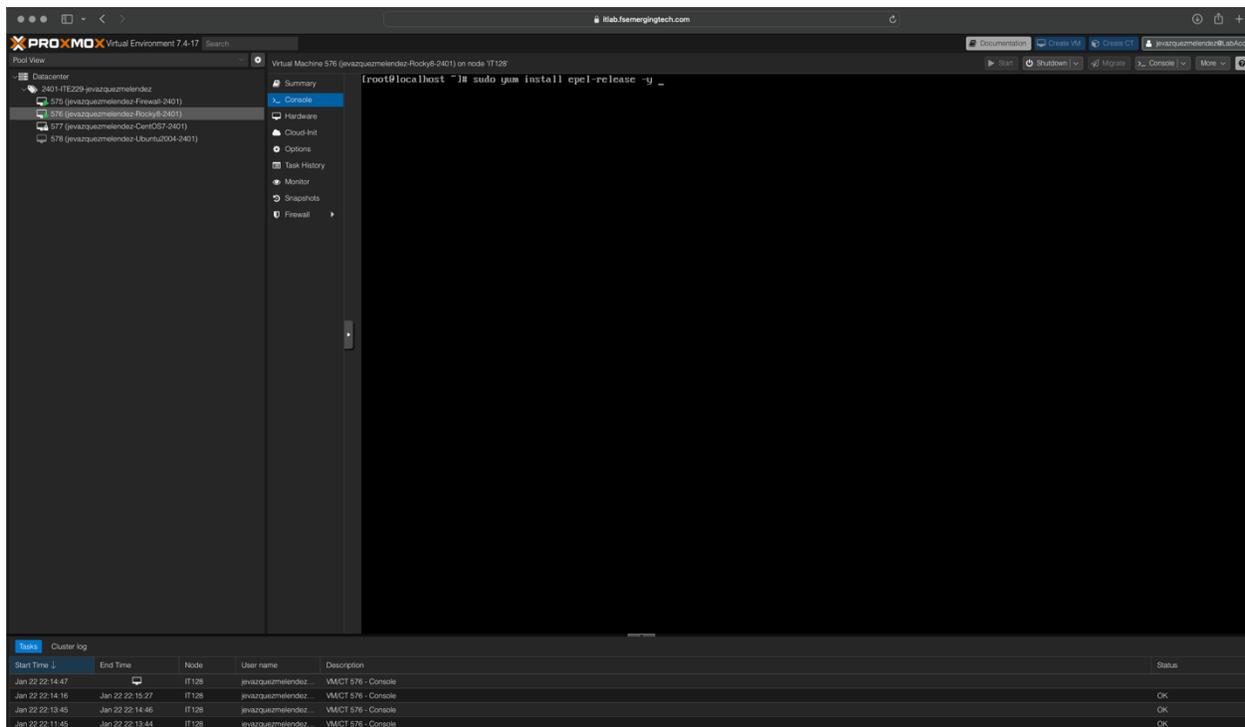
The screenshot shows the Proxmox VE interface. On the left, a tree view shows the cluster structure with three VMs: VMCT 576 (Rocky8-2401), VMCT 577 (CentOS-2401), and VMCT 578 (Ubuntu-2401). The 'Console' tab for VMCT 576 is selected. The main window displays a terminal session with the prompt `[root@localhost ~]# sudo yum update -y_`. At the bottom, a 'Tasks' table shows the execution of the console command.

Start Time	End Time	Node	User name	Description	Status
Jan 22 22:06:17		IT128	juvazquzmelendez	VMCT 576 - Console	
Jan 22 22:06:07	Jan 22 22:06:16	IT127	juvazquzmelendez	VMCT 577 - Console	OK
Jan 22 22:06:07	Jan 22 22:06:07	IT128	juvazquzmelendez	VMCT 578 - Console	OK

Install EPEL Packages

```
# sudo yum install epel-release -y
```

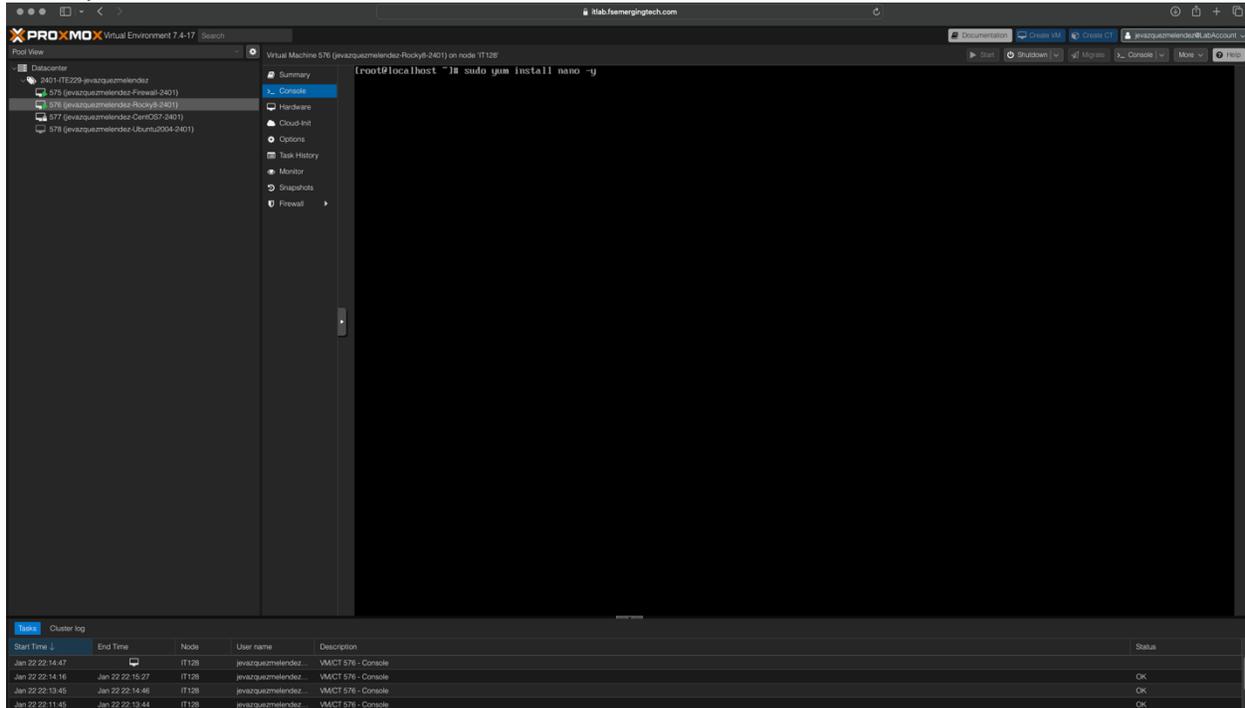
The command `sudo yum install epel-release` installs your system's Extra Packages for Enterprise Linux (EPEL) repository. This repository contains additional software packages not included in the default repositories provided by your Linux distribution.



Install Nano Editor

```
# sudo yum install nano -y
```

The command `sudo yum install nano` installs the Nano text editor on your system. Nano is a simple, user-friendly text editor that can create and edit text files.



Disable SELinux

```
# sudo nano /etc/selinux/config
```

To disable SELinux by editing its configuration file, you can follow these steps:

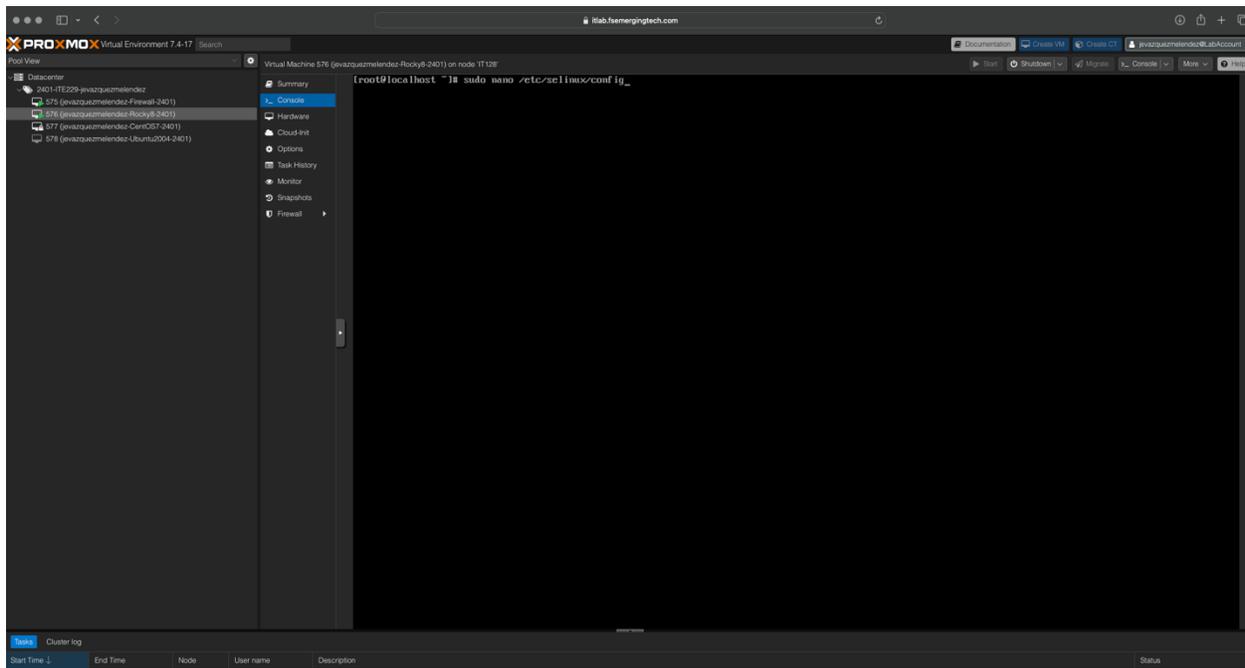
Run the following command to open the SELinux configuration file in the Nano text editor with root privileges:

```
# sudo nano /etc/selinux/config
```

Locate the line that starts with SELINUX= and change its value to disabled.

Save the changes by pressing Ctrl + X and pressing enter.

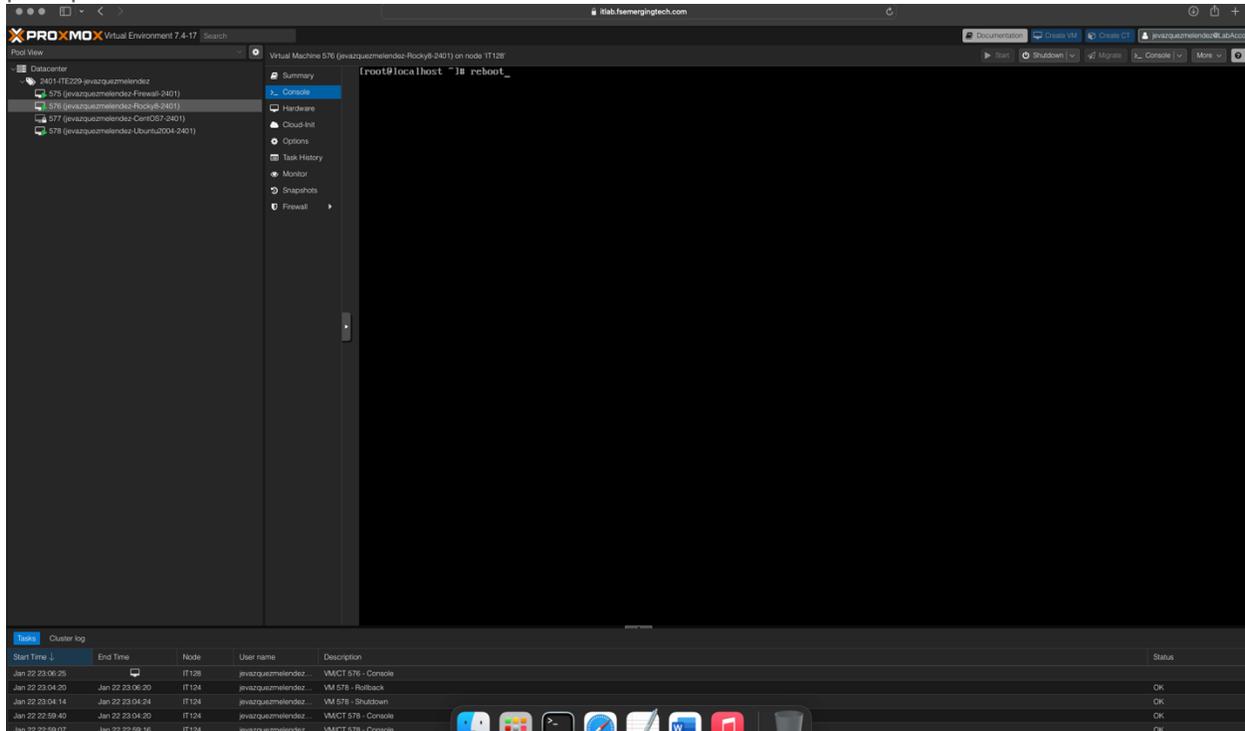
Reboot your system for the changes to take effect.



Reboot VM

reboot

This command is used to reboot or reset a Virtual Machine from the terminal shell using a command prompt



The screenshot displays the Proxmox VE interface. The main window shows a terminal session for VM 576 (jevazquzmelendez-Rocky8-2401) on node IT128. The terminal prompt is `[root@localhost ~]#` and the command `reboot_` has been entered. The sidebar on the left contains navigation options: Summary, Hardware, Cloud-Init, Options, Task History, Monitor, Snapshots, and Firewall. At the bottom, a task log table is visible.

Start Time ↓	End Time	Node	User name	Description	Status
Jan 22 23:06:25		IT128	jevazquzmelendez	VMCT 576 - Console	
Jan 22 23:04:20	Jan 22 23:06:20	IT124	jevazquzmelendez	VM 578 - Rollback	OK
Jan 22 23:04:14	Jan 22 23:04:24	IT124	jevazquzmelendez	VM 578 - Shutdown	OK
Jan 22 22:59:40	Jan 22 23:04:20	IT124	jevazquzmelendez	VMCT 578 - Console	OK
Jan 22 22:59:07	Jan 22 22:59:16	IT124	jevazquzmelendez	VMCT 578 - Console	OK

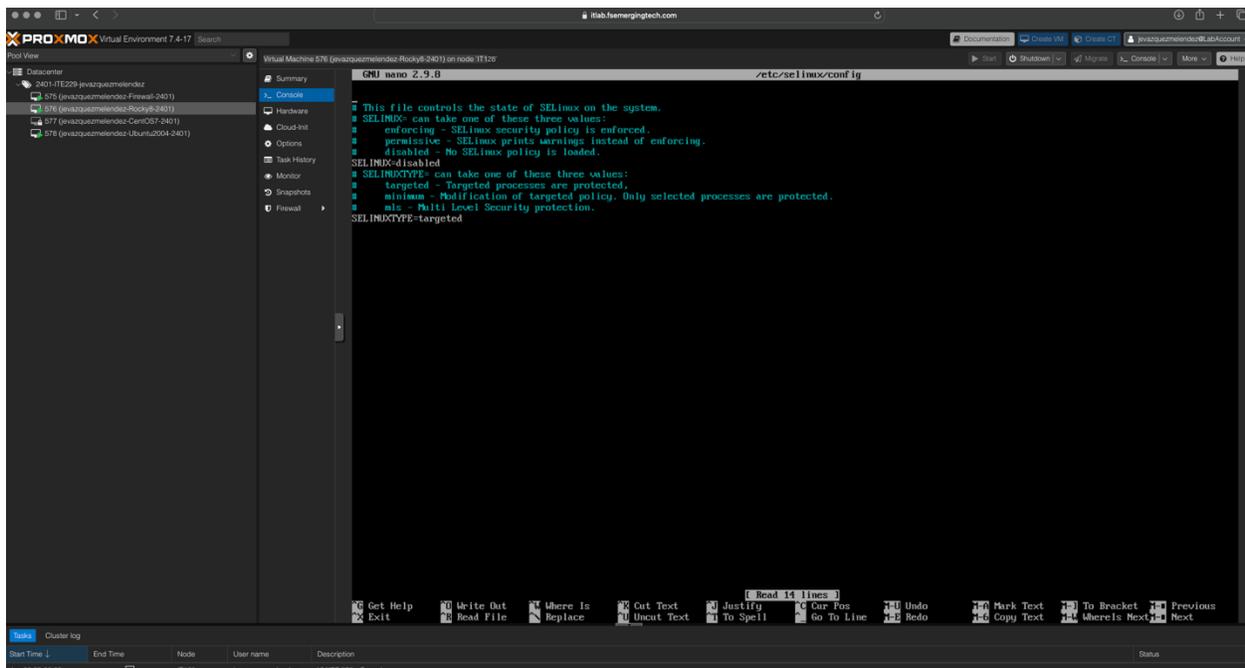
Test SELinux

```
# sudo nano /etc/selinux/config
```

Locate the line that starts with SELINUX= and change its value to disabled.

Save the changes by pressing Ctrl + X and pressing enter.

Reboot your system for the changes to take effect.

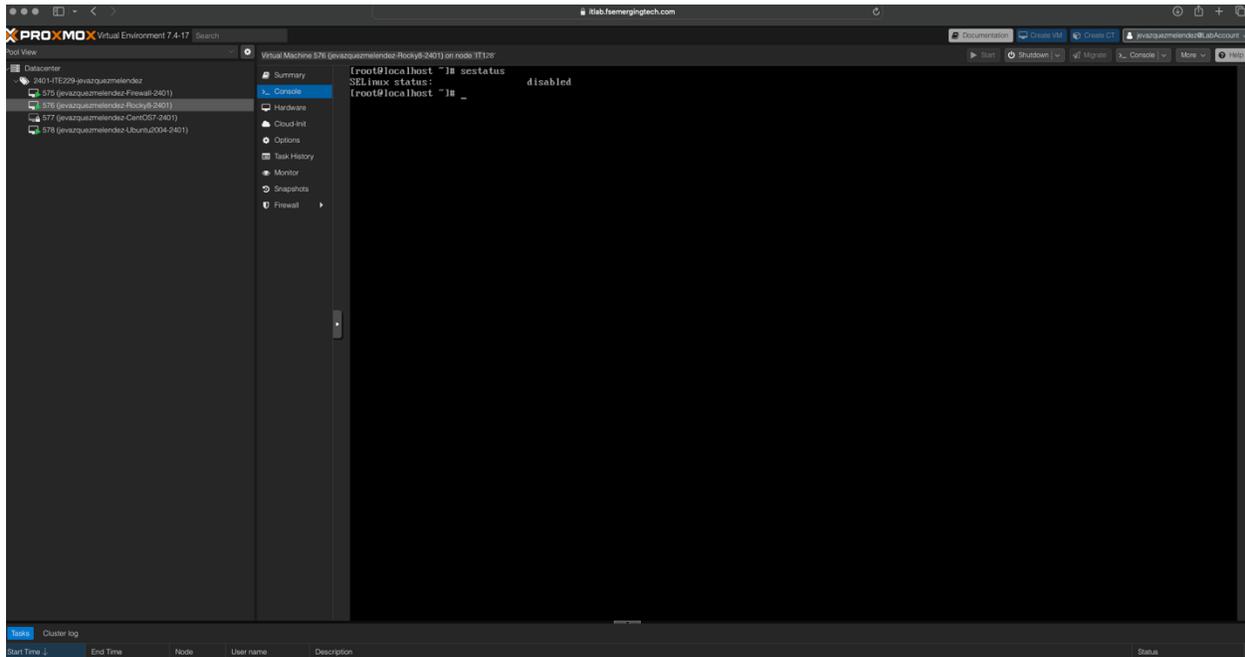


```
PROXMOX Virtual Environment 7.4-17
Virtual Machine 576 (proxmox/rook-2401) on node 1T12Z
GNU nano 2.9.8 /etc/selinux/config
# This file controls the state of SELinux on the system.
# SELINUX can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - No SELinux policy is loaded.
SELINUX=disabled
# SELINUXTYPE can take one of these three values:
#   targeted - Targeted processes are protected.
#   minimum - Modification of targeted policy. Only selected processes are protected.
#   mlse - Multi Level Security protection.
SELINUXTYPE=targeted
```

Confirm SELinux Status

sestatus

This command is used to check the status of the SELinux (Security-Enhanced Linux) system on a Linux machine.

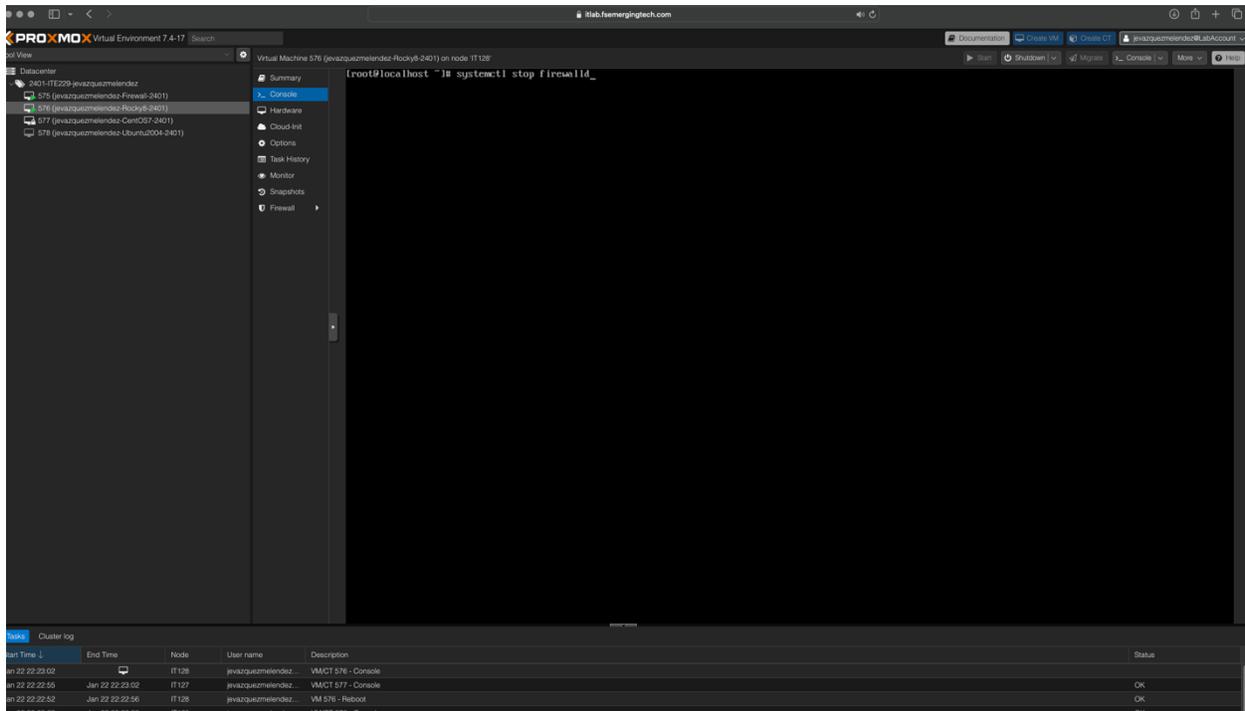


Rocky Firewall

Stop Firewall

```
# systemctl stop firewalld
```

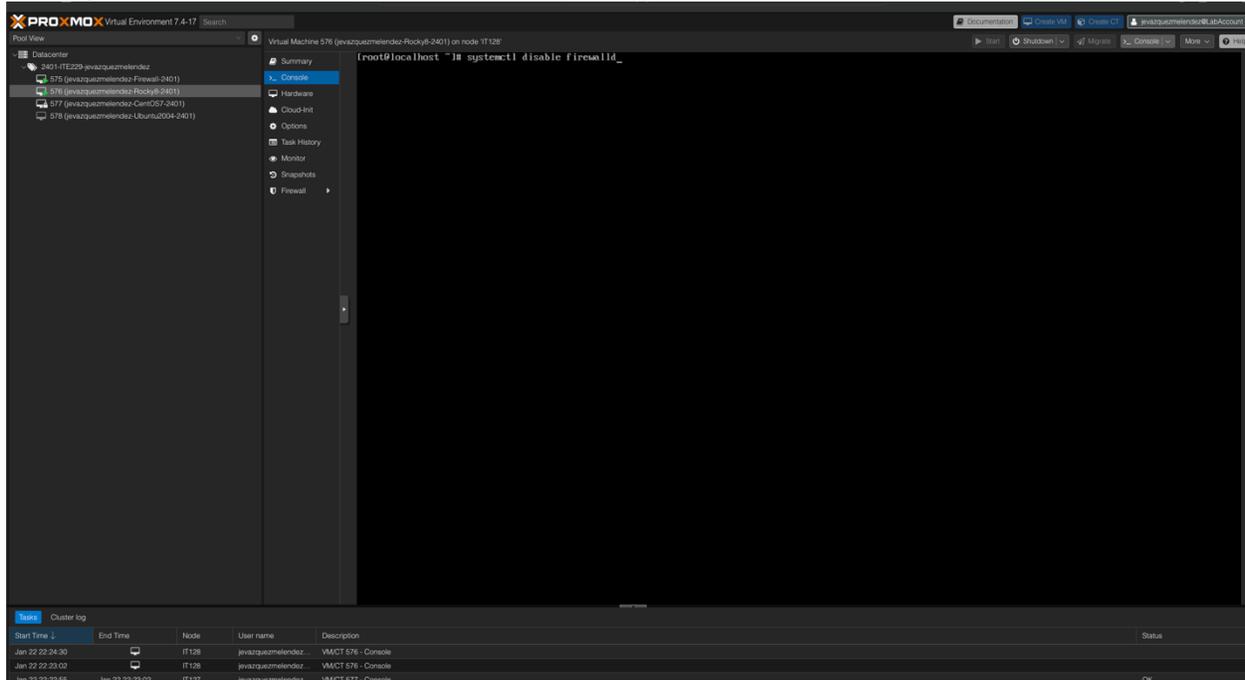
The `systemctl stop firewalld` command is used to stop the firewall service on your Linux system. When you stop the firewall service, all network traffic to and from your system will be allowed by default.



Disable Firewall

```
# systemctl disable firewalld
```

On the other hand, the `systemctl disable firewalld` command is used to prevent the firewall service from starting at boot time. Even if you restart your system, the firewall service will not start automatically.

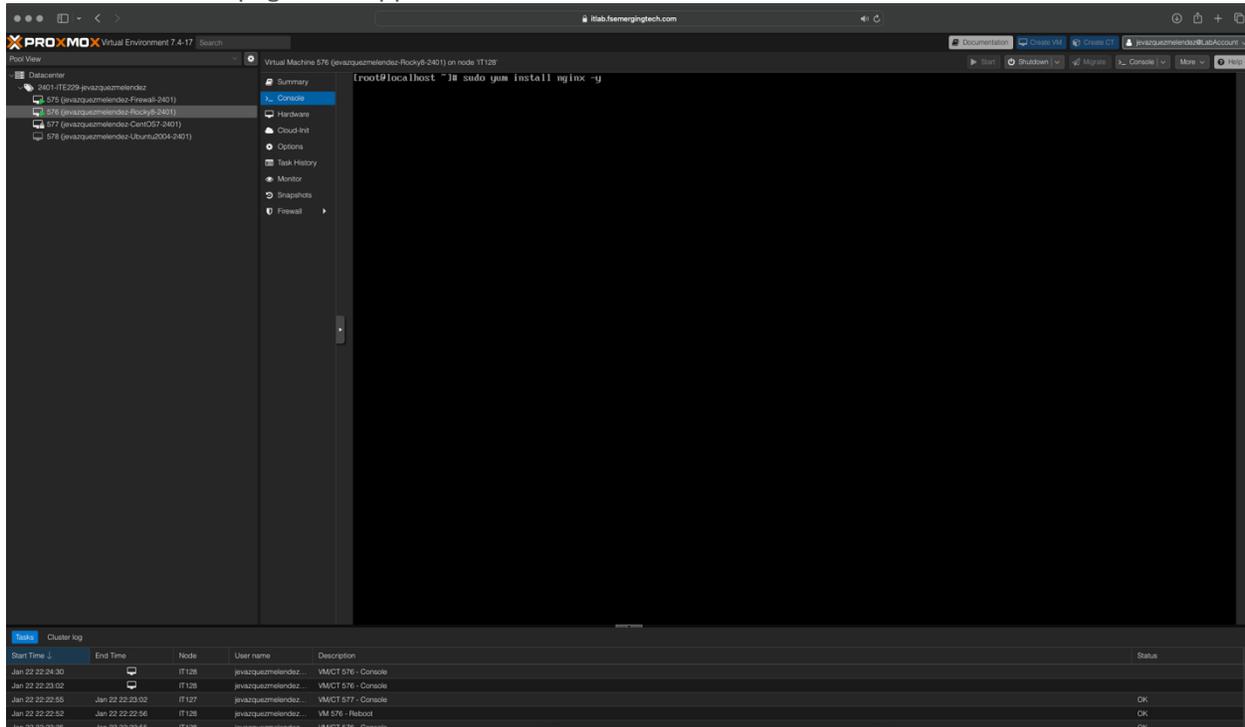


NginX

Install NginX

```
# sudo yum install nginx -y
```

This command installs the Nginx web server on the system. Nginx is a popular open-source web server that can serve web pages and applications

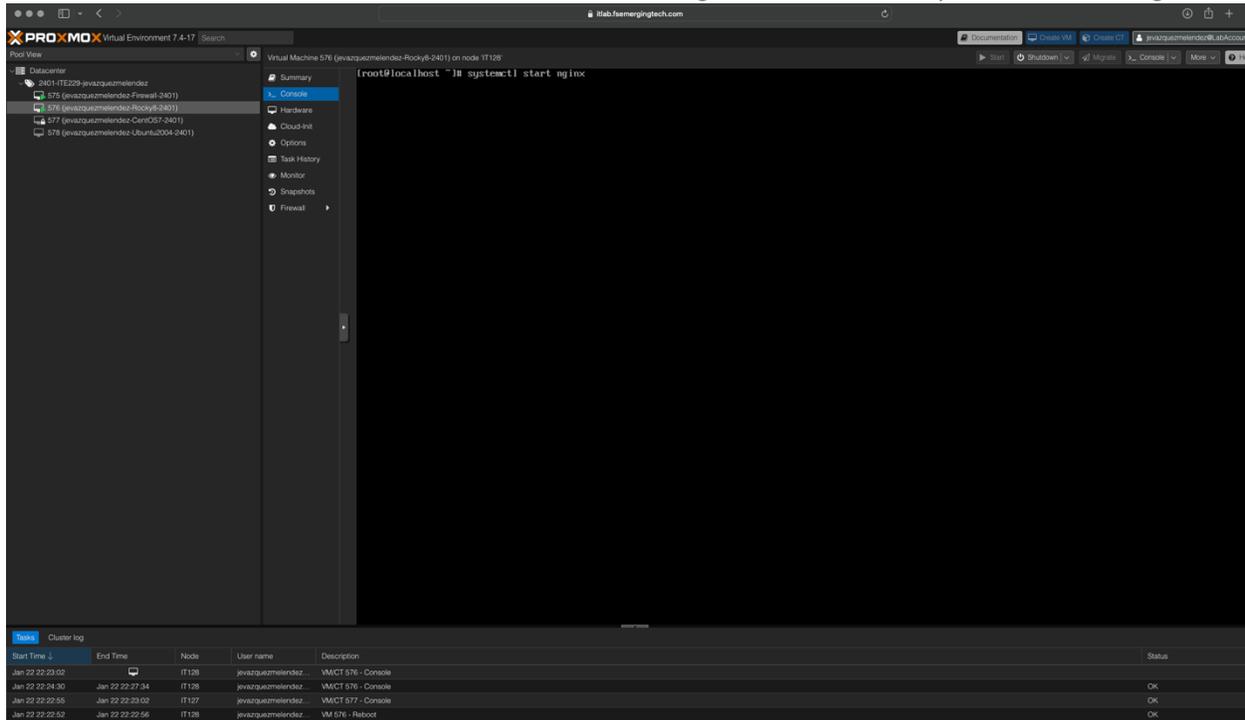


Start NginX

```
# systemctl start nginx
```

This command starts the Nginx web server service on your Linux system.

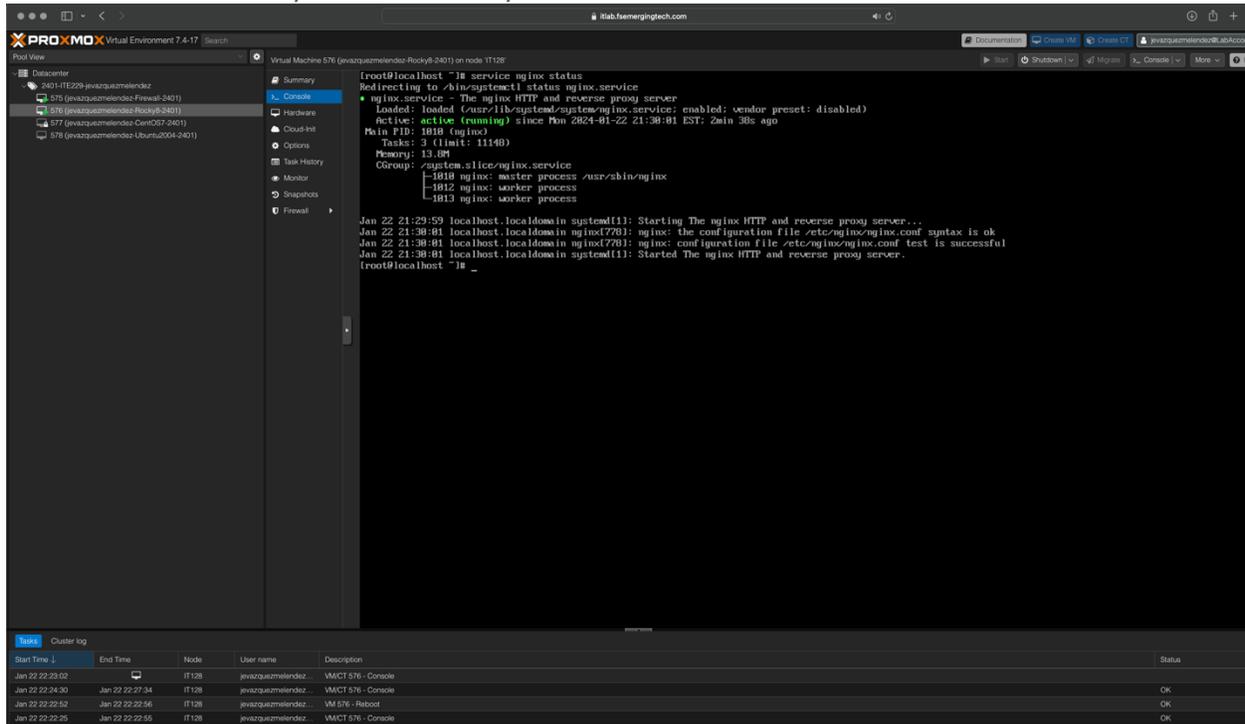
This command initiates the Nginx service, and the web server will start serving web pages and applications. It is an effortless command that can be used to start the Nginx service manually if it is not running.



Enable NginX

```
# systemctl enable nginx
```

This command enables the Nginx service to start automatically at boot time. This means the Nginx service will automatically start when the system is started or restarted.



The screenshot shows a terminal window in a Proxmox VE environment. The user is logged in as root on localhost. The terminal output shows the following commands and their results:

```
[root@localhost ~]# service nginx status
Redirecting to /bin/systemctl status nginx.service
● nginx.service - The nginx HTTP and reverse proxy server
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; vendor preset: disabled)
   Active: active (running) since Mon 2024-01-22 21:38:01 EST; 2min 38s ago
     Main PID: 1810 (nginx)
           Tasks: 3 (limit: 11140)
        Memory: 13.0M
           CGroup: /system.slice/nginx.service
                  └─1810 nginx: master process /usr/sbin/nginx
                    └─1812 nginx: worker process
                      └─1813 nginx: worker process

Jan 22 21:29:59 localhost.localdomain systemd[1]: Starting The nginx HTTP and reverse proxy server...
Jan 22 21:38:01 localhost.localdomain nginx[770]: nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
Jan 22 21:38:01 localhost.localdomain nginx[770]: nginx: configuration file /etc/nginx/nginx.conf test is successful
Jan 22 21:38:01 localhost.localdomain systemd[1]: Started The nginx HTTP and reverse proxy server.
[root@localhost ~]#
```

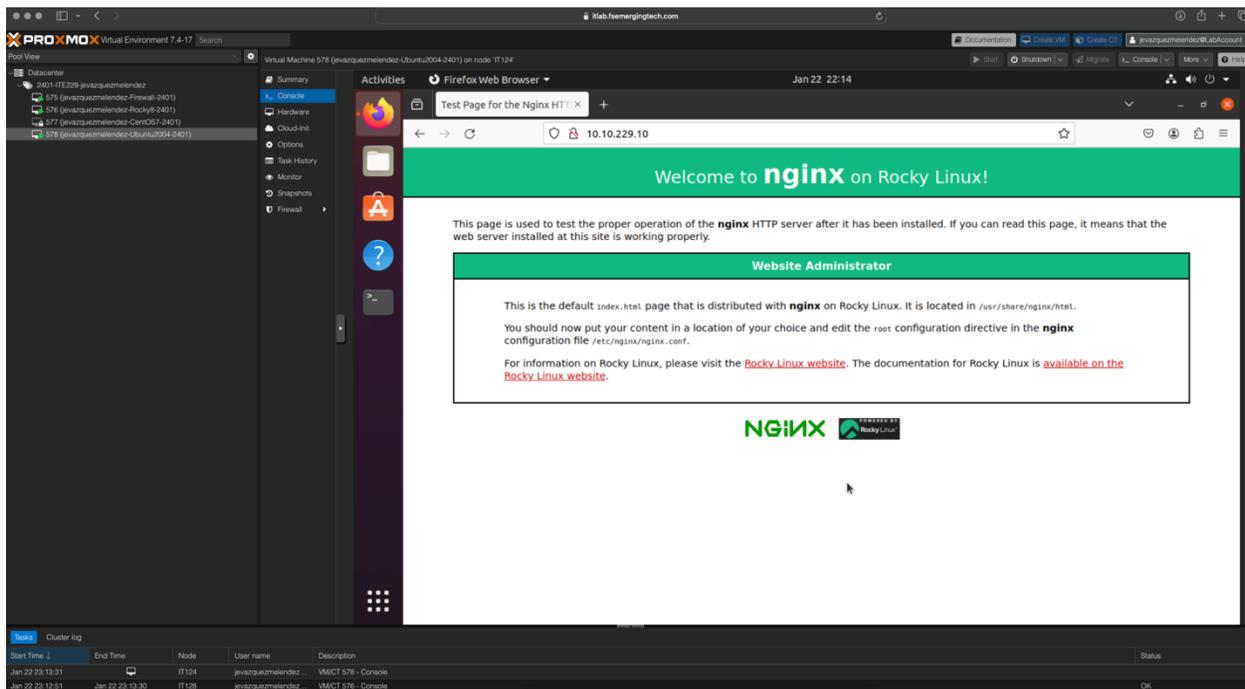
Below the terminal window, there is a table showing the cluster log entries:

Start Time	End Time	Node	User name	Description	Status
Jan 22 22:23:02		IT128	javaquemeleendez...	VMCT 576 - Console	
Jan 22 22:24:30	Jan 22 22:27:34	IT128	javaquemeleendez...	VMCT 576 - Console	OK
Jan 22 22:22:52	Jan 22 22:22:56	IT128	javaquemeleendez...	VM 576 - Reboot	OK
Jan 22 22:22:25	Jan 22 22:22:56	IT128	javaquemeleendez...	VMCT 576 - Console	OK

Test NginX

This command starts the Nginx web server service on your Linux system. This command initiates the Nginx service, and the web server will start serving web pages and applications. It is an effortless command that can be used to start the Nginx service manually if it is not running.

To test a Nginx server with a Ghost Docker container running on a specific IP address (let's say IP address 10.10.229.10, you can use a web browser to access the IP address of the Nginx server from the Ubuntu VM. If the Nginx server is appropriately configured to proxy requests to the Ghost container, you should see the Ghost blog's homepage when accessing the IP address of the Nginx server through the web browser.

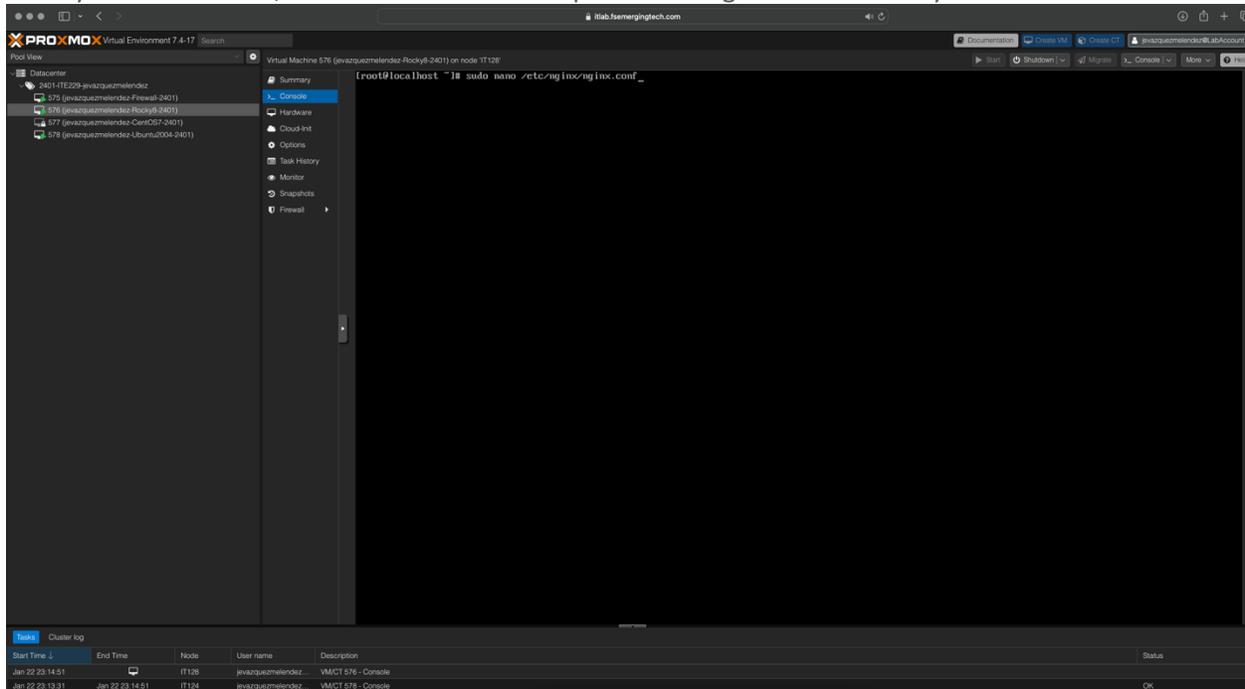


Reverse Proxy for Ghost Site

Edit NginX configuration file

```
# sudo nano /etc/nginx/nginx.conf
```

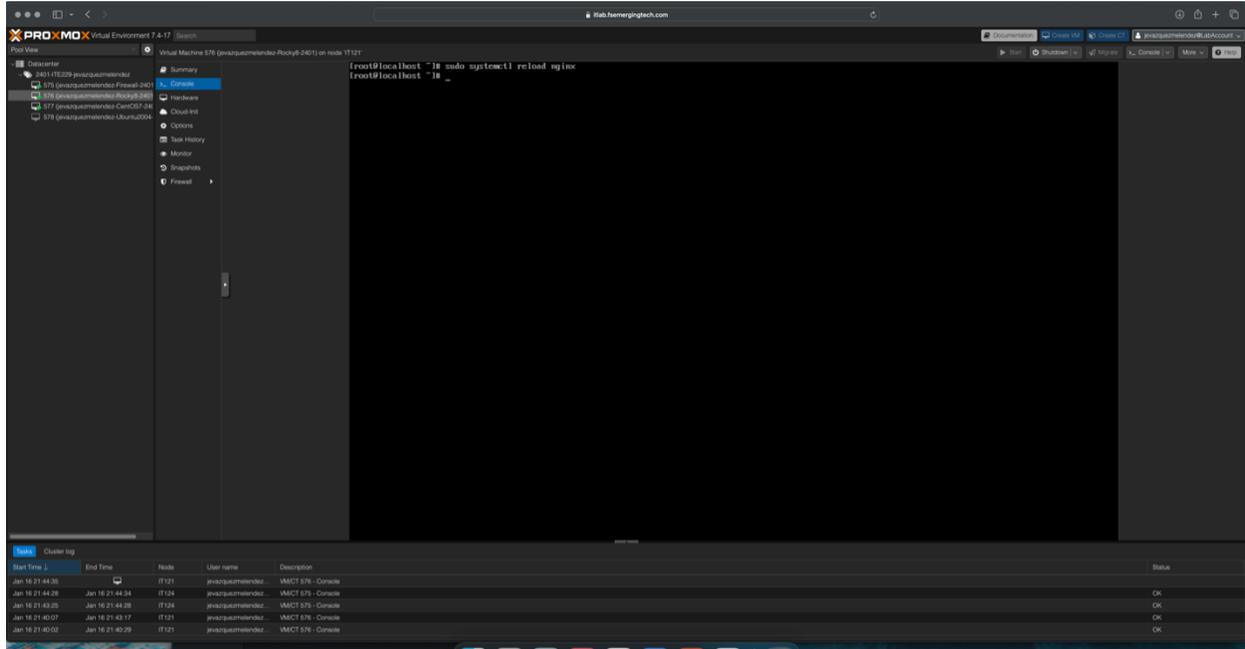
Editing the Nginx configuration file is a way to make changes to the settings of a web server, such as Ghost. Modifying this file allows you to adjust the server's behavior, performance, security, and more. The process involves opening the file, finding the relevant section, making changes, and saving the file. Once you've done that, the server will use the updated configuration to serve your content.



Reload NginX service

```
# sudo systemctl reload nginx
```

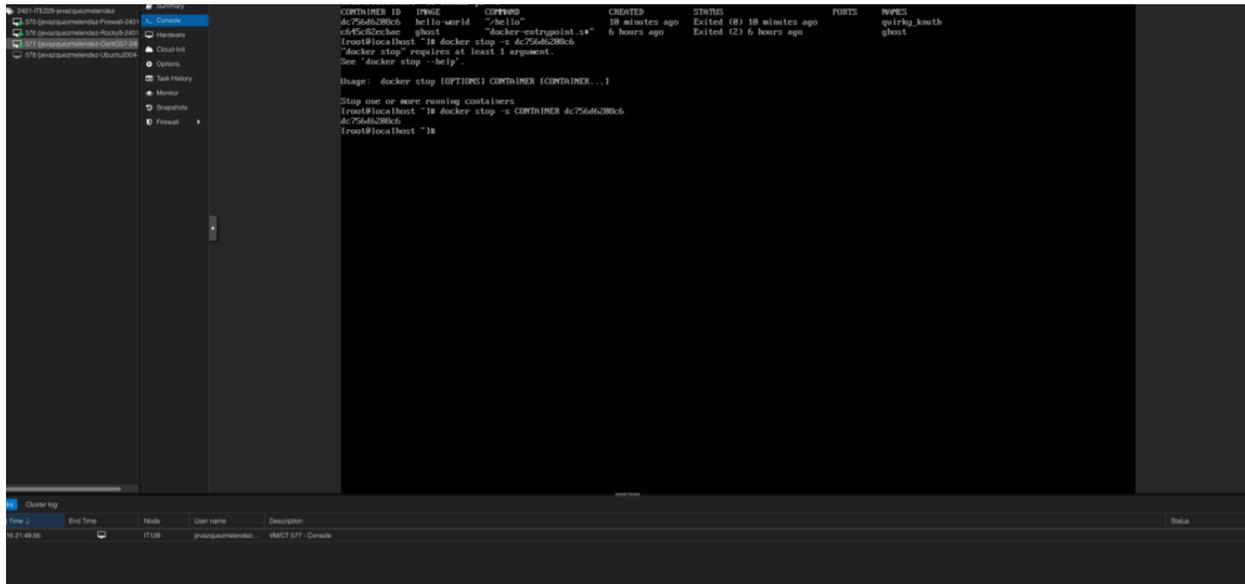
The command you entered will reload the configuration of the nginx service using sudo privileges. This is useful if you have made changes to the nginx configuration files and want those changes to take effect without restarting the entire service.



Terminate Docker

```
# docker stop -s CONTAINER [CONTAINER-ID]
```

This command stops a running Docker container. The "-s" option is used to specify the signal to be sent to the container when it is stopped. In this case, "CONTAINER" is just a placeholder for the actual name or ID of the container that needs to be stopped. So, when this command is executed, the specified container will be gracefully stopped, and all the processes running inside the container will be terminated.



```
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS          NAMES
4c7566b288c6   hello-world    "hello"                 18 minutes ago   Exited (0)   19 minutes ago
445c50e20cbe   ghost         "docker-entrypoint.sh"  6 hours ago     Exited (2)   6 hours ago

[root@localhost ~]# docker stop -s 4c7566b288c6
"docker stop" requires at least 1 argument.
See "docker stop --help".

Usage:  docker stop [OPTIONS] CONTAINER [CONTAINER...]

Stop one or more running containers.
[root@localhost ~]# docker stop -s CONTAINER 4c7566b288c6
4c7566b288c6
[root@localhost ~]#
```

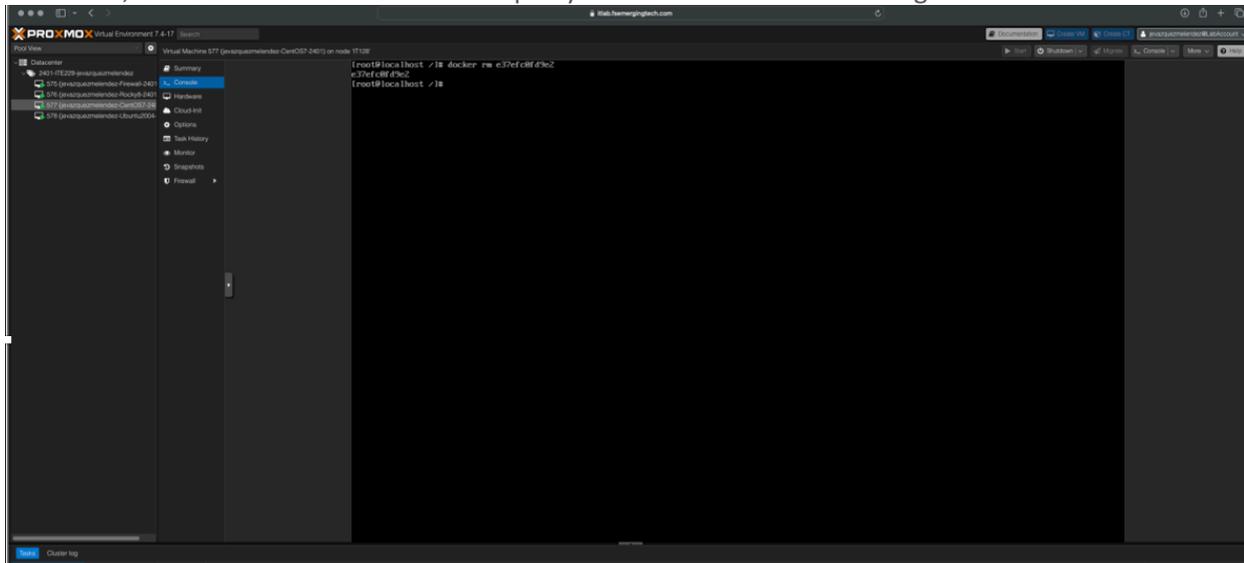
Cluster log

Time ↓	End Time	Node	User Name	Description	State
16:21:49.56		IT128	jenasquimendez...	VMCT 577 - Console	

Delete Ghost Container

```
# docker rm [CONTAINER-ID]
```

The "docker rm [CONTAINER-ID]" command removes a stopped or running Docker container. The "[CONTAINER-ID]" part of the command should be replaced with the actual ID of the container that you want to remove. If the specified container runs, the command will stop it before removing it. It's important to note that any data or changes made inside the container will be lost after the container is removed. Therefore, it's recommended to first backup any critical data before removing the container



Create New Ghost Container

```
# docker run -d --name ghost -p 3001:2368 -e url=http://10.10.229.10/blog ghost
```

This command is used in Docker, a popular software platform for creating and managing virtual containers.

The command "docker run" is used to start a new container. In this case, the container is named "ghost".

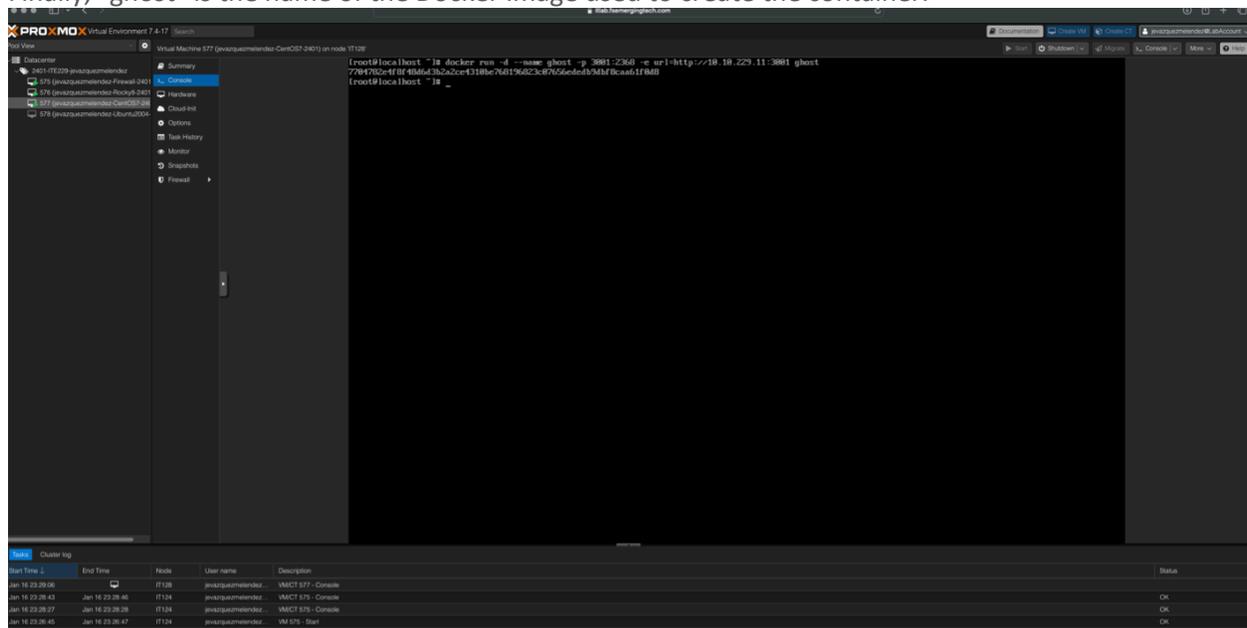
The "-d" flag runs the container in the background as a daemon.

The "--name" flag gives the container a specific name, in this case, "ghost."

The "-p" flag maps a port between the container and the host. It maps port 2368 in the container to port 3001 on the host computer.

The "-e" flag is used to set an environment variable. In this case, the environment variable "url" is set to <http://10.10.229.10/blog>.

Finally, "ghost" is the name of the Docker image used to create the container.

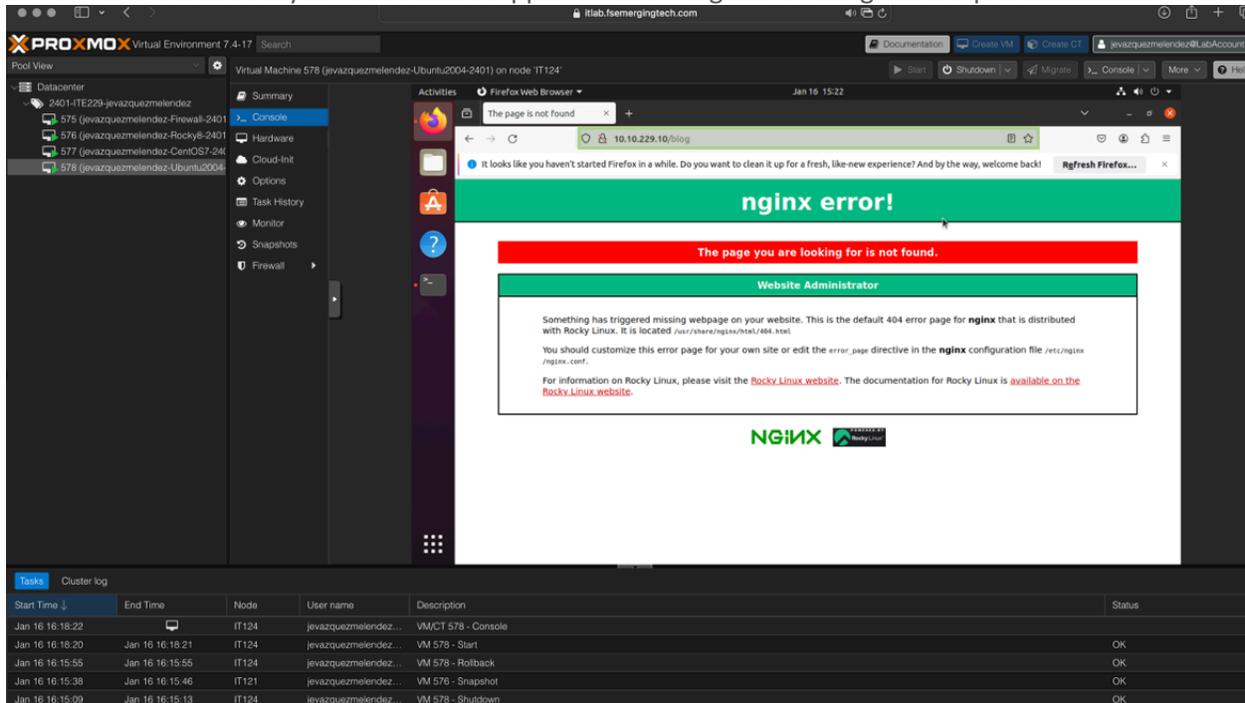


Browse to Ghost

To browse Nginx from an Ubuntu machine in simple terms, you can follow these steps:
Open a web browser on your Ubuntu machine.

In the address bar, type the IP address or hostname of the server where Nginx is running.
<http://10.10.229.10>

You can now browse any website or web application that Nginx is serving on that port.



The screenshot shows a Proxmox Virtual Environment interface. The main window displays a Firefox web browser with the address bar set to `10.10.229.10/blog`. The browser shows a 404 error page with the following content:

nginx error!

The page you are looking for is not found.

Website Administrator

Something has triggered missing webpage on your website. This is the default 404 error page for **nginx** that is distributed with Rocky Linux. It is located `/usr/share/nginx/html/404.html`.

You should customize this error page for your own site or edit the `error_page` directive in the **nginx** configuration file `/etc/nginx/nginx.conf`.

For information on Rocky Linux, please visit the [Rocky Linux website](#). The documentation for Rocky Linux is [available on the Rocky Linux website](#).

NGINX

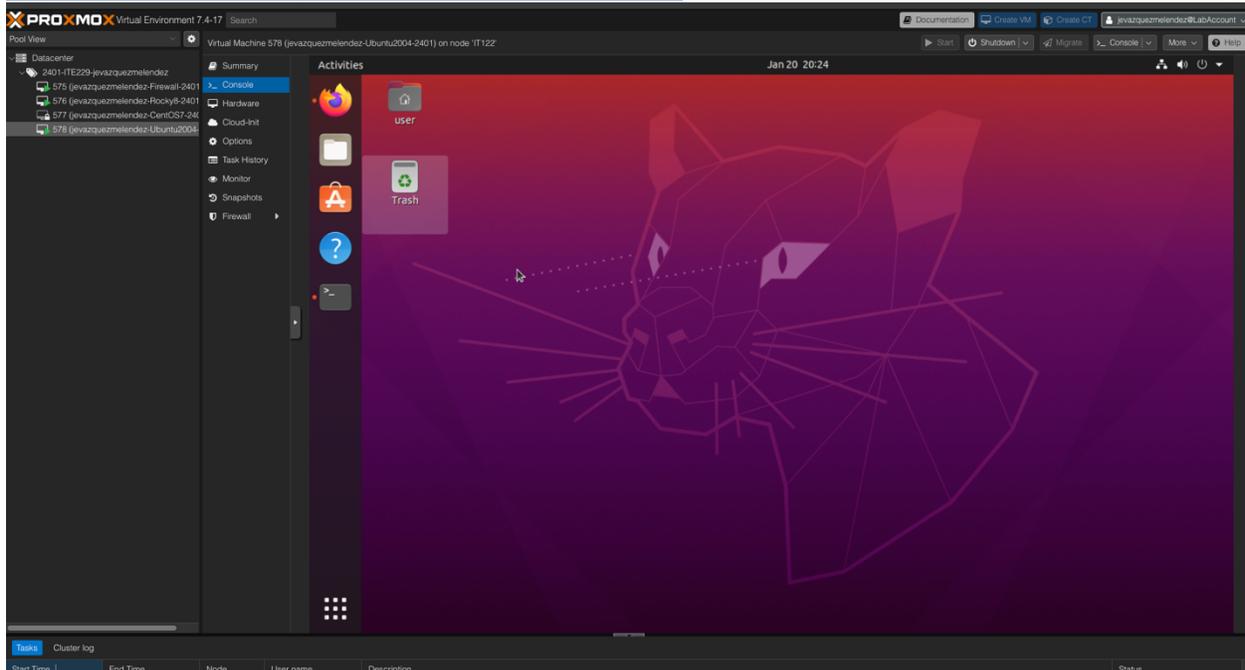
Below the browser window, a 'Tasks' table is visible, showing a cluster log with the following data:

Start Time	End Time	Node	User name	Description	Status
Jan 16 16:18:22		IT124	jevazquezmelendez...	VMCT 578 - Console	
Jan 16 16:18:20	Jan 16 16:18:21	IT124	jevazquezmelendez...	VM 578 - Start	OK
Jan 16 16:15:55	Jan 16 16:15:55	IT124	jevazquezmelendez...	VM 578 - Rollback	OK
Jan 16 16:15:38	Jan 16 16:15:46	IT121	jevazquezmelendez...	VM 576 - Snapshot	OK
Jan 16 16:15:09	Jan 16 16:15:13	IT124	jevazquezmelendez...	VM 578 - Shutdown	OK

IMPORTANT: To earn full credit, original formatting of this document must remain in place. In addition, ALL screenshots must include a “full view”, including your ProxMox title bar with your username showing.

Milestone #2: WordPress on Ubuntu - LAMP Stack

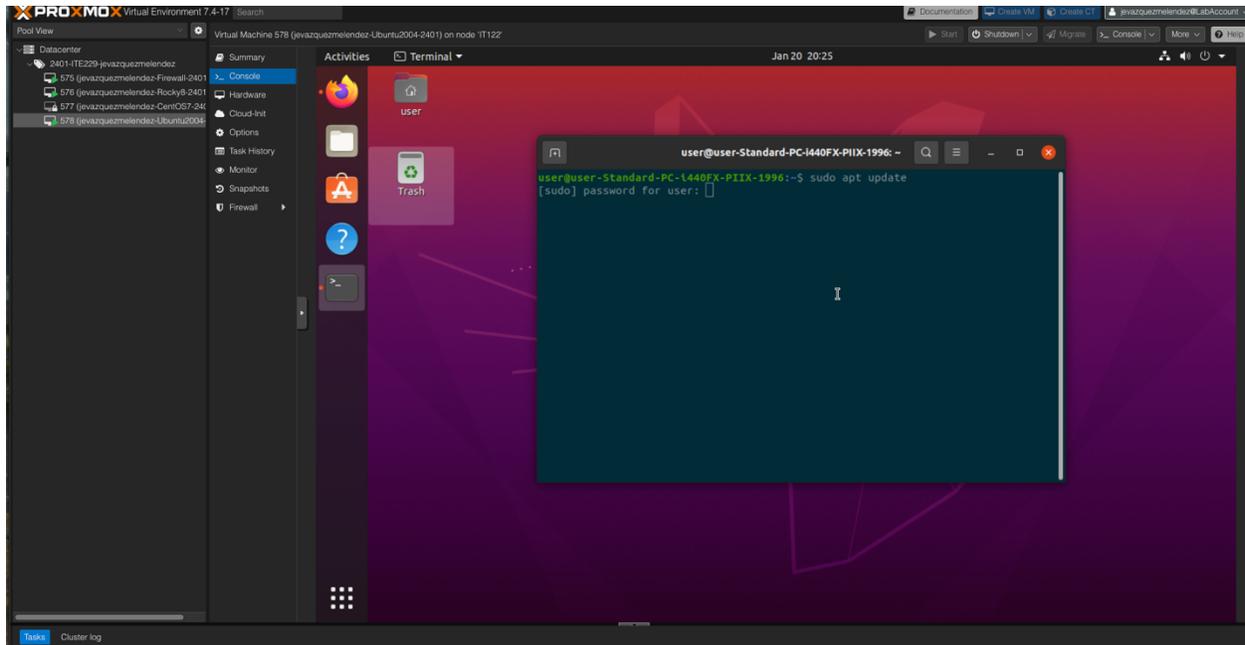
Show screenshot of your Ubuntu Console in VE



Update Ubuntu

```
$ sudo apt update -y
```

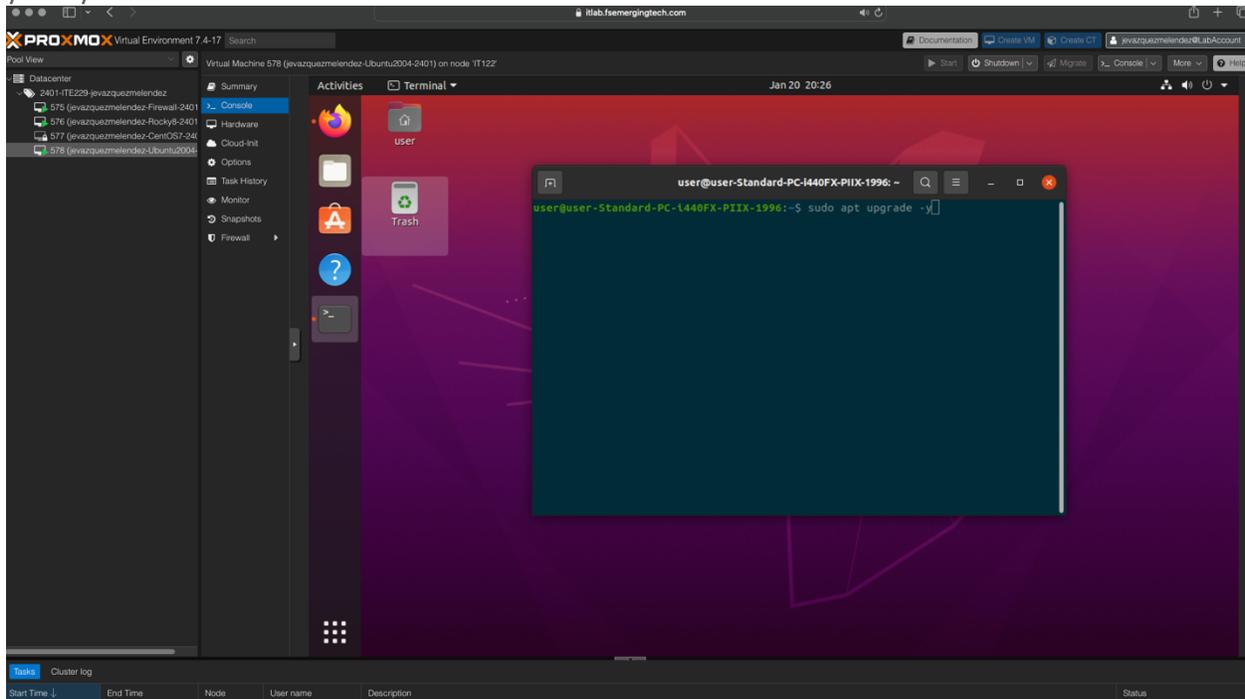
This command is used to update the list of available software packages from the internet on your Ubuntu system.



Upgrade Ubuntu

```
$ sudo apt upgrade -y
```

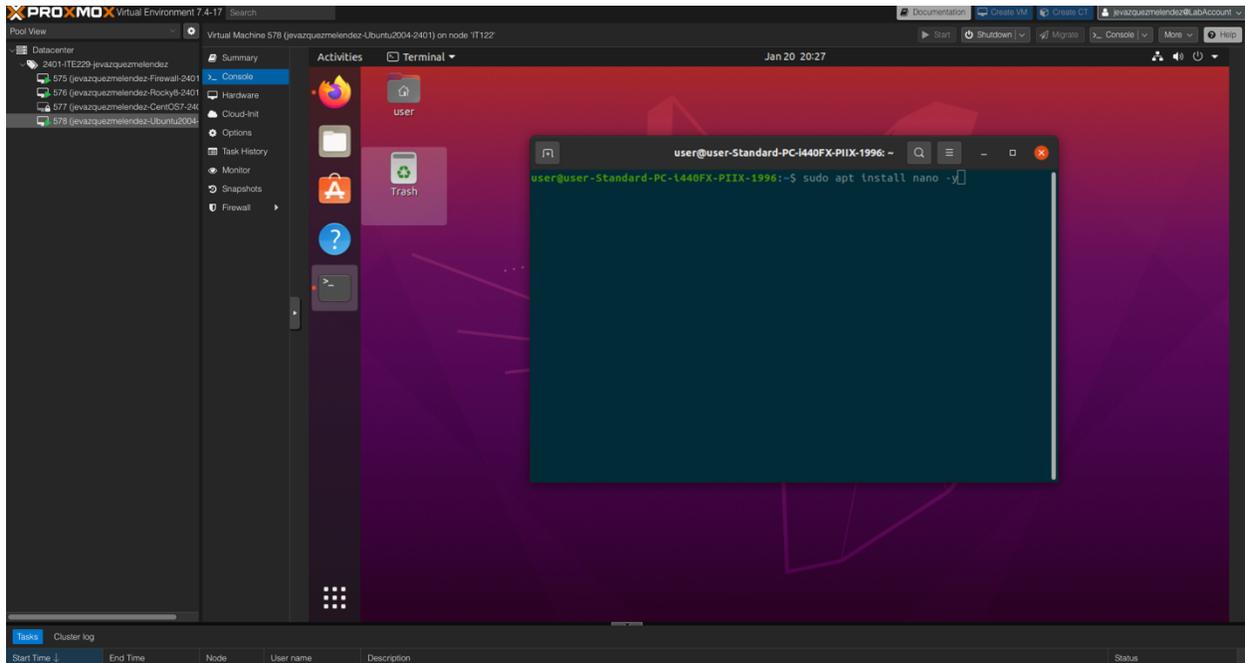
This command is used to download and install any available updates for the installed software packages on your system.



Install Nano Editor

```
$ sudo apt install nano -y
```

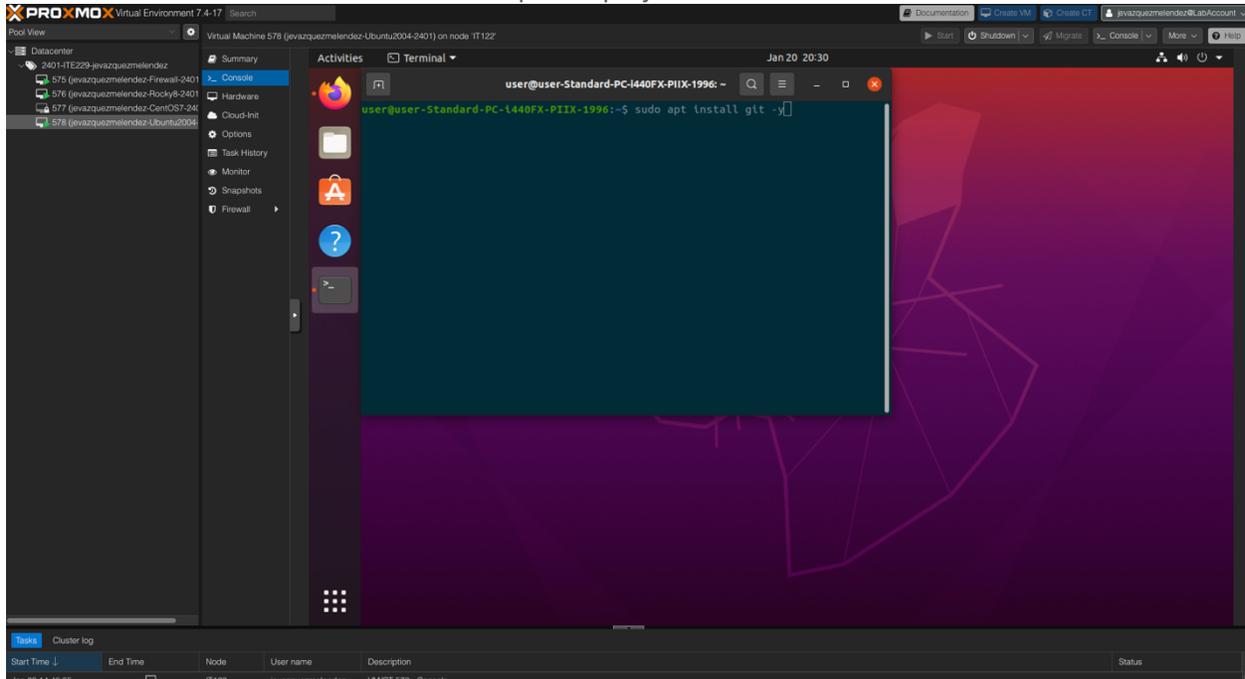
This command is used to install a text editor called 'nano' on your Ubuntu system, which can be used to modify files.



Install Git

```
$ sudo apt install git -y
```

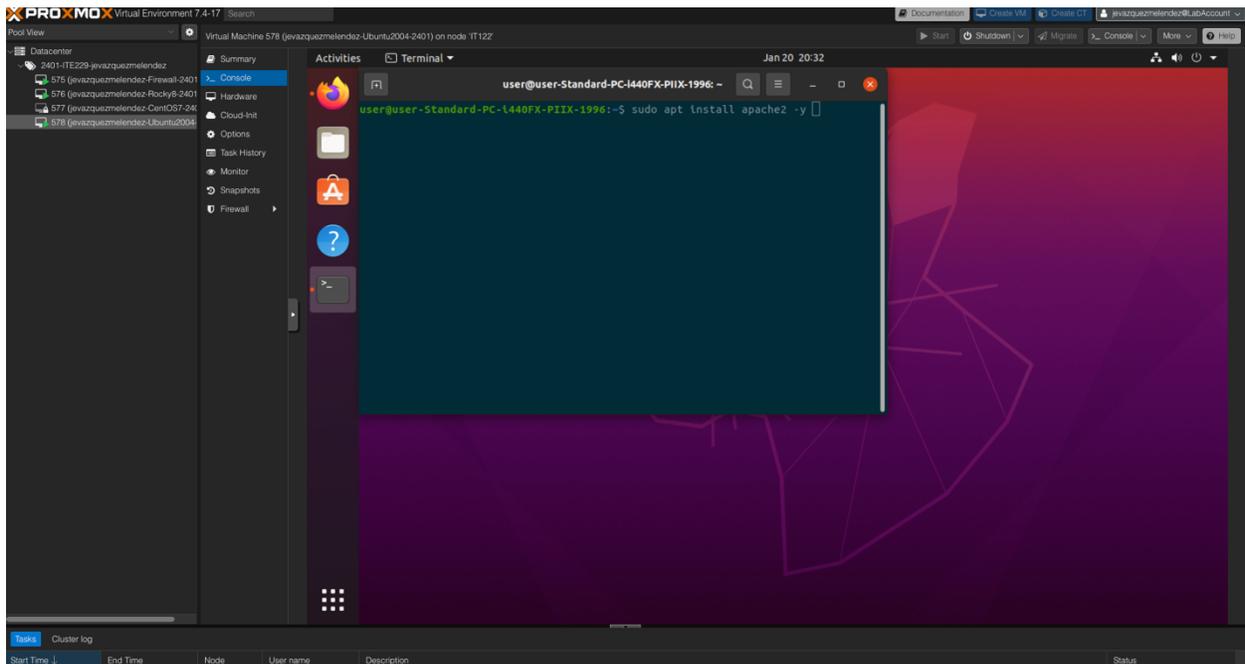
This command is used to install Git, a version control system that allows you to track changes in files and collaborate with others on software development projects called GitHub.



Install Apache2

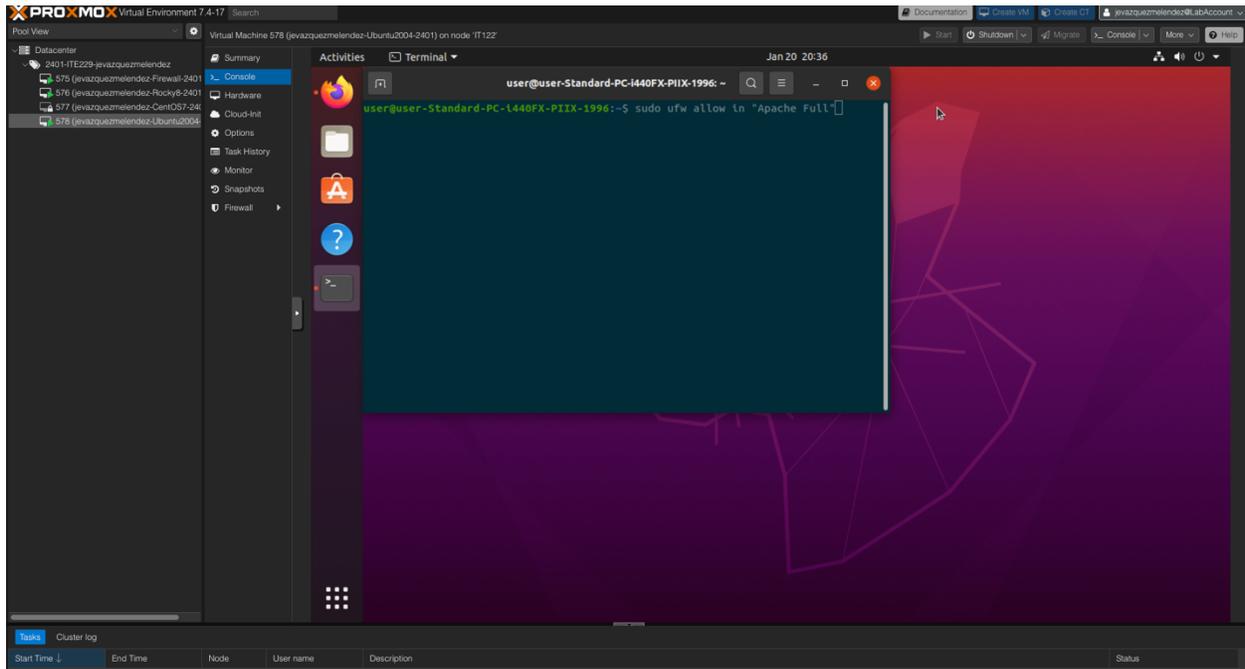
```
$ sudo apt install apache2 -y
```

The command `sudo apt install apache2` is used to install the Apache web server on your Ubuntu machine. This will allow you to host the project website or web application on your computer or virtual machine.



Open Firewall Ports 80 and 443
\$ sudo ufw allow in "Apache Full"

The command is used to allow incoming traffic to your Apache web server through the firewall. This is important because by default, the firewall blocks all incoming traffic to your computer. The 'Apache Full' option allows access to both HTTP and HTTPS traffic.



Browse to Apache2 Ubuntu Default Page

To Browser to the Apache2 Ubuntu Default Page you are going to open the Firefox Web Browser and Type:

<http://10.10.229.12>

It should redirect you to this page.

The screenshot shows a Proxmox Virtual Environment (VE) interface. The main window displays a Firefox Web Browser with the Apache2 Ubuntu Default Page loaded at the IP address 10.10.229.12. The page features the Ubuntu logo and the title "Apache2 Ubuntu Default Page". Below the title, there is a red banner that says "It works!". The main content of the page explains that this is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It also provides a "Configuration Overview" section, which includes a code block showing the default configuration files for Apache2 on Ubuntu:

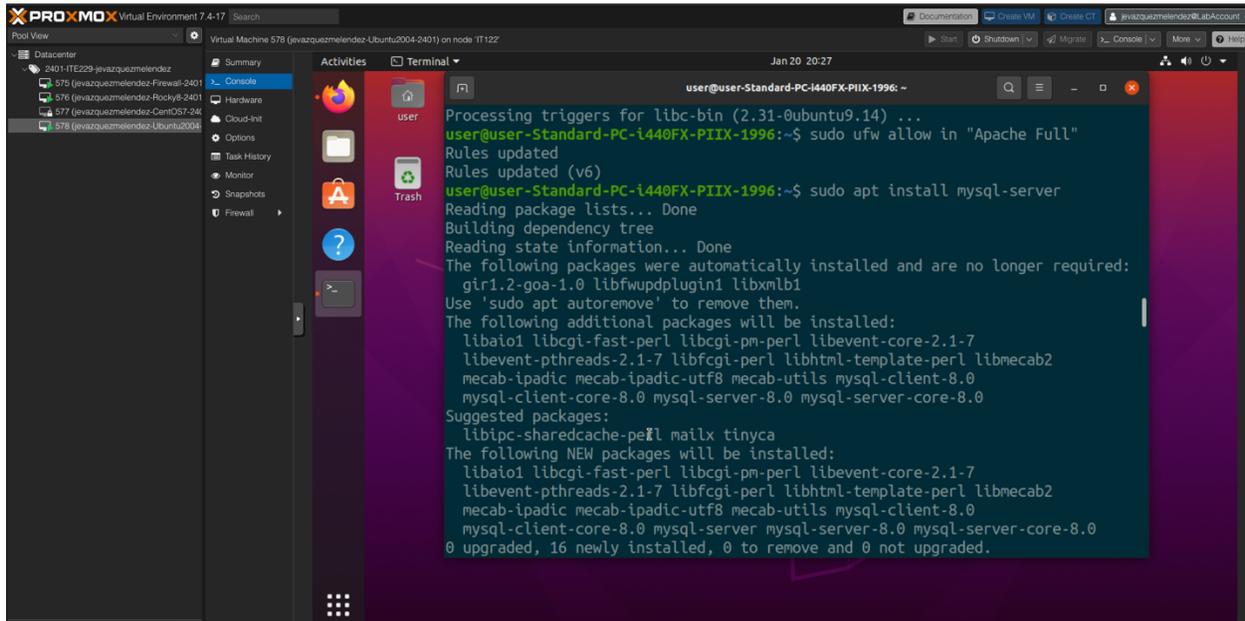
```
etc/apache2/  
|-- apache2.conf  
|   |-- ports.conf  
|-- mods-enabled  
|   |-- *.load  
|   |-- *.conf  
|-- conf-enabled  
|   |-- *.conf  
|-- sites-enabled  
|   |-- *.conf
```

The Proxmox interface also shows a sidebar with various system settings and a task log at the bottom.

Install MySQL

```
$ sudo apt install mysql-server
```

This command allows you to install the first command, install the MySQL database server on your computer or virtual machine.



The screenshot shows a Proxmox VE interface with a terminal window open. The terminal output shows the installation of MySQL server. The user first runs `sudo ufw allow in "Apache Full"` to update firewall rules. Then, they run `sudo apt install mysql-server`, which lists additional packages to be installed and suggests some packages to be removed. The terminal output is as follows:

```
Processing triggers for libc-bin (2.31-0ubuntu9.14) ...
user@user-Standard-PC-i440FX-PIIX-1996:~$ sudo ufw allow in "Apache Full"
Rules updated
Rules updated (v6)
user@user-Standard-PC-i440FX-PIIX-1996:~$ sudo apt install mysql-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
gir1.2-goa-1.0 libfwupdplugin1 libxmlb1
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
libaio1 libbcgl-fast-perl libbcgl-pm-perl libevent-core-2.1-7
libevent-pthreads-2.1-7 libfcgi-perl libhtml-template-perl libmecab2
mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0
mysql-client-core-8.0 mysql-server-8.0 mysql-server-core-8.0
Suggested packages:
libipc-sharedcache-perl mailx tinycsa
The following NEW packages will be installed:
libaio1 libbcgl-fast-perl libbcgl-pm-perl libevent-core-2.1-7
libevent-pthreads-2.1-7 libfcgi-perl libhtml-template-perl libmecab2
mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0
mysql-client-core-8.0 mysql-server mysql-server-8.0 mysql-server-core-8.0
0 upgraded, 16 newly installed, 0 to remove and 0 not upgraded.
```

Alter root user password

TDJFid34c935kg59DR

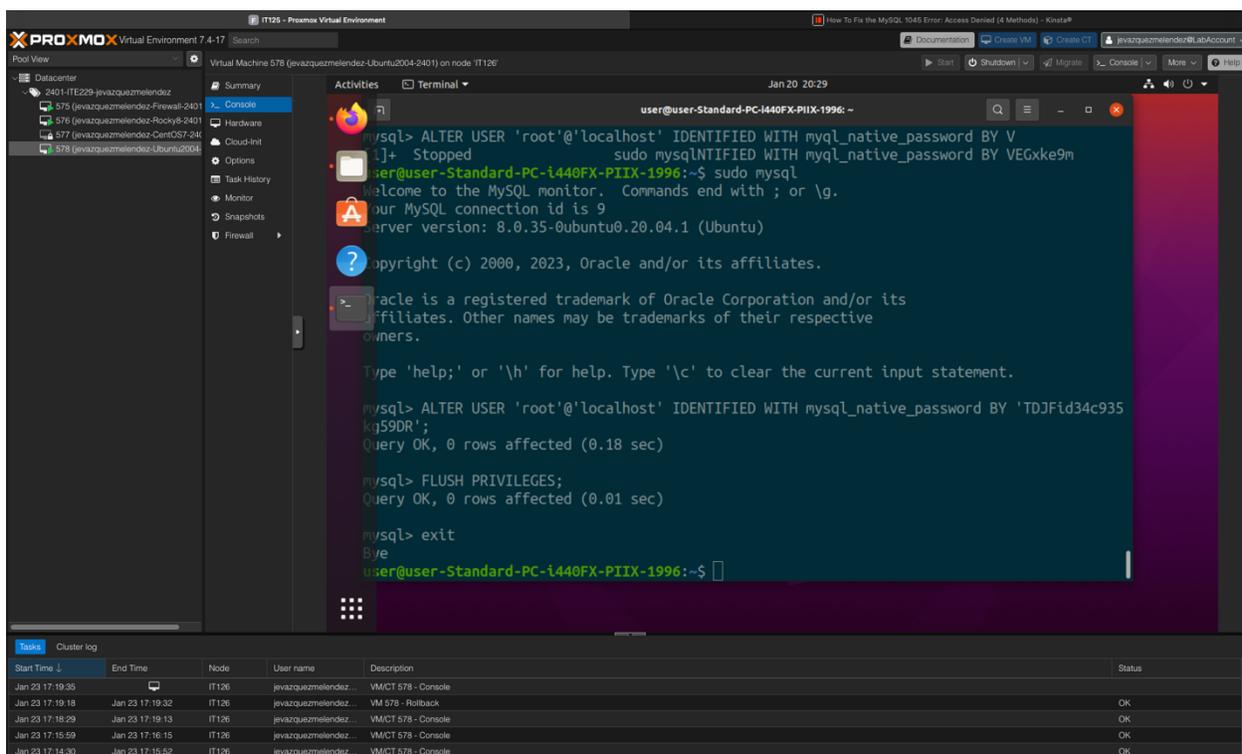
```
$ sudo mysql
```

This allows you to run the mysql command as the root user, which grants you more permissions and access to the MySQL server using the sudo privileges.

Then to Alter root user password type the following command:

```
mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'TDJFid34c935kg59DR';
```

Then Press Enter.



The screenshot shows a terminal window within a Proxmox virtual environment. The terminal prompt is `user@user-Standard-PC-l440FX-PIIX-1996:~`. The user has executed `sudo mysql`, which has opened the MySQL command-line interface. The MySQL prompt is `mysql>`. The user has entered the command `ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'TDJFid34c935kg59DR';`, which has been executed successfully, returning `Query OK, 0 rows affected (0.18 sec)`. The user has then entered `FLUSH PRIVILEGES;`, returning `Query OK, 0 rows affected (0.01 sec)`, and finally `exit`, returning `Bye`. The terminal prompt is now `user@user-Standard-PC-l440FX-PIIX-1996:~$`.

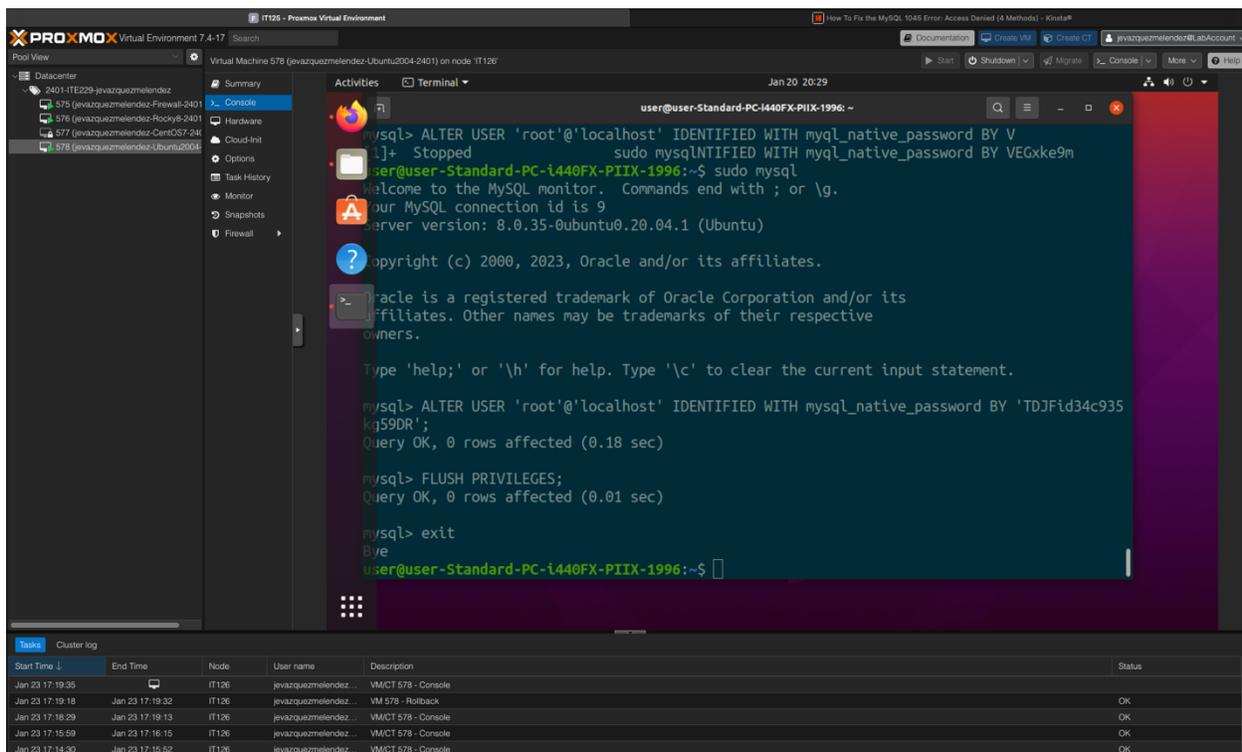
Start Time	End Time	Node	User name	Description	Status
Jan 23 17:19:35		IT126	jevaqzsmelendez...	VMCT 578 - Console	
Jan 23 17:19:18	Jan 23 17:19:32	IT126	jevaqzsmelendez...	VM 578 - Rollback	OK
Jan 23 17:18:29	Jan 23 17:19:13	IT126	jevaqzsmelendez...	VMCT 578 - Console	OK
Jan 23 17:15:59	Jan 23 17:16:15	IT126	jevaqzsmelendez...	VMCT 578 - Console	OK
Jan 23 17:14:30	Jan 23 17:15:52	IT126	jevaqzsmelendez...	VMCT 578 - Console	OK

Flush Privileges

```
mysql> FLUSH PRIVILEGES;
```

Press Enter.

The command FLUSH PRIVILEGES; is used in MySQL to reload the grant tables and refresh the user privileges. This is useful when you have to make changes to the MySQL database user accounts or privileges and want to ensure that the changes take effect immediately.



```
user@user-Standard-PC-l440FX-PIIX-1996:~$ sudo mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.35-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY '
]'+ Stopped
user@user-Standard-PC-l440FX-PIIX-1996:~$ sudo mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.35-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'TDJFid34c935
kg59DR';
Query OK, 0 rows affected (0.18 sec)

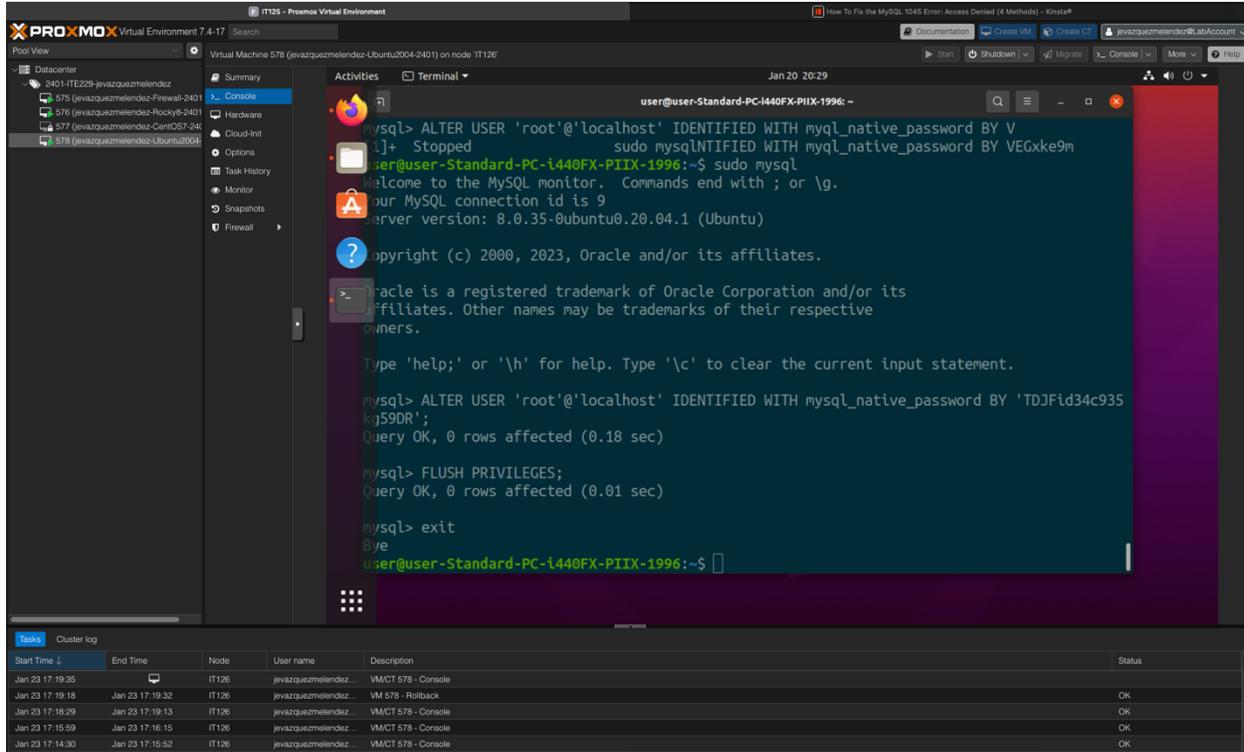
mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.01 sec)

mysql> exit
Bye
user@user-Standard-PC-l440FX-PIIX-1996:~$
```

Start Time	End Time	Node	User name	Description	Status
Jan 23 17:19:35		IT126	jevaqzqmlelendez	VMCT 578 - Console	
Jan 23 17:19:18	Jan 23 17:19:32	IT126	jevaqzqmlelendez	VM 578 - Rollback	OK
Jan 23 17:18:29	Jan 23 17:19:13	IT126	jevaqzqmlelendez	VMCT 578 - Console	OK
Jan 23 17:15:59	Jan 23 17:16:15	IT126	jevaqzqmlelendez	VMCT 578 - Console	OK
Jan 23 17:14:30	Jan 23 17:15:52	IT126	jevaqzqmlelendez	VMCT 578 - Console	OK

Quit MySQL
mysql> exit

The exit command is used to quit the MySQL command-line interface. When you type this command and press Enter, the MySQL shell will close, and you will return to the ubuntu shell.



The screenshot shows a Proxmox VE terminal window with the following content:

```
user@user-Standard-PC-i440FX-PIIX-1996:~$ sudo mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.35-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY '
kq59DR';
Query OK, 0 rows affected (0.18 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.01 sec)

mysql> exit
Bye
user@user-Standard-PC-i440FX-PIIX-1996:~$
```

Below the terminal window, there is a 'Tasks' table with the following data:

Start Time	End Time	Node	User name	Description	Status
Jan 23 17:19:25		IT126	jevaquezmelendez	VMCT 578 - Console	
Jan 23 17:19:18	Jan 23 17:19:32	IT126	jevaquezmelendez	VM 578 - Rollback	OK
Jan 23 17:18:29	Jan 23 17:19:13	IT126	jevaquezmelendez	VMCT 578 - Console	OK
Jan 23 17:15:59	Jan 23 17:16:15	IT126	jevaquezmelendez	VMCT 578 - Console	OK
Jan 23 17:14:30	Jan 23 17:15:52	IT126	jevaquezmelendez	VMCT 578 - Console	OK

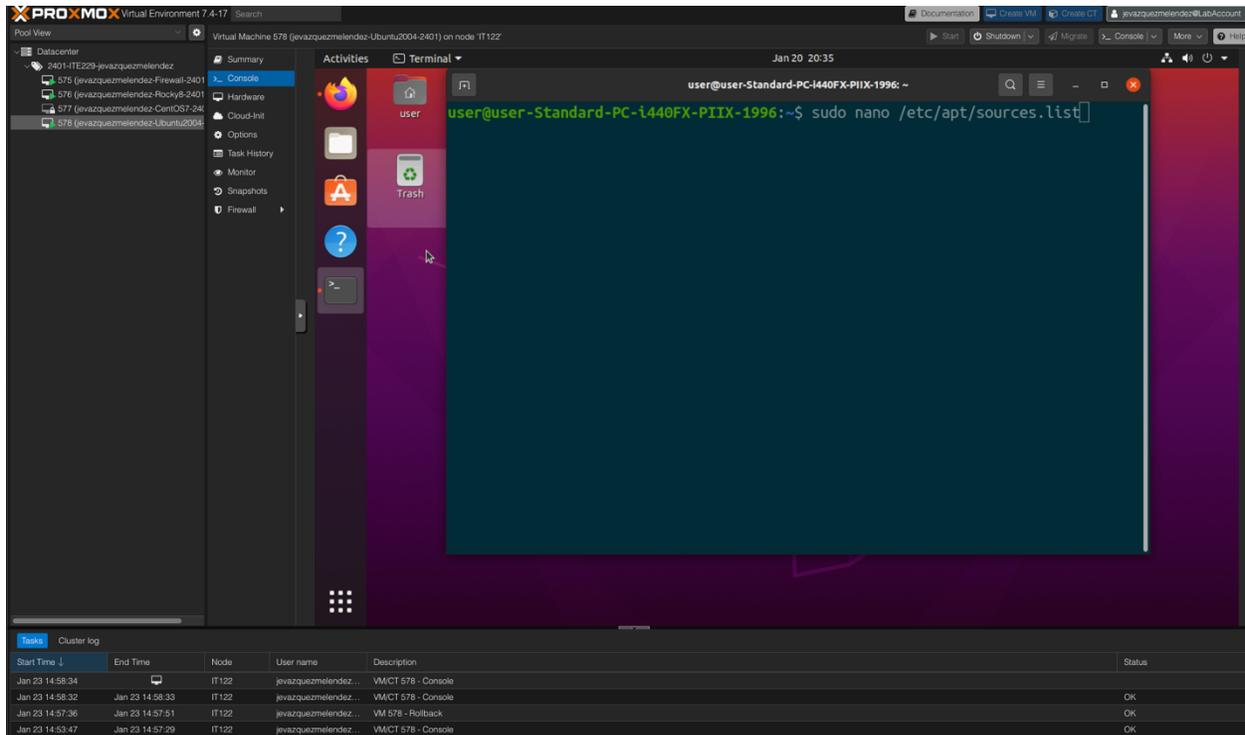
Install PHP

Edit Sources.list File

```
$ sudo nano /etc/apt/sources.list
```

To put it simply, you need to open the sources. The list file is located at /etc/apt/ using the nano text editor with sudo privileges.

Then, you need to add the "universe" repository to all the URLs listed in the file.



PROXMO Virtual Environment 7.4-17

Virtual Machine 578 (jevaqzqmellendez-Ubuntu2004-2401) on node: IT122

Jan 20 20:37

user@user-Standard-PC-I440FX-PIIX-1996: -

```

GNU nano 4.8 /etc/apt/sources.list Modified
## Uncomment the following two lines to add software from Canonical's
## 'partner' repository.
## This software is not part of Ubuntu, but is offered by Canonical and the
## respective vendors as a service to Ubuntu users.
# deb http://archive.canonical.com/ubuntu focal partner
# deb-src http://archive.canonical.com/ubuntu focal partner

deb http://security.ubuntu.com/ubuntu focal-security main restricted universe
# deb-src http://security.ubuntu.com/ubuntu focal-security main restricted
deb http://security.ubuntu.com/ubuntu focal-security universe
# deb-src http://security.ubuntu.com/ubuntu focal-security universe
deb http://security.ubuntu.com/ubuntu focal-security multiverse universe
# deb-src http://security.ubuntu.com/ubuntu focal-security multiverse

# This system was installed using small removable media
# (e.g. netinst, live or single CD). The matching "deb cdrom"
# entries were disabled at the end of the installation process.
# For information about how to configure apt package sources,
# see the sources.list(5) manual.

Save modified buffer?
Y Yes
N No      ^C Cancel
  
```

Tasks Cluster log

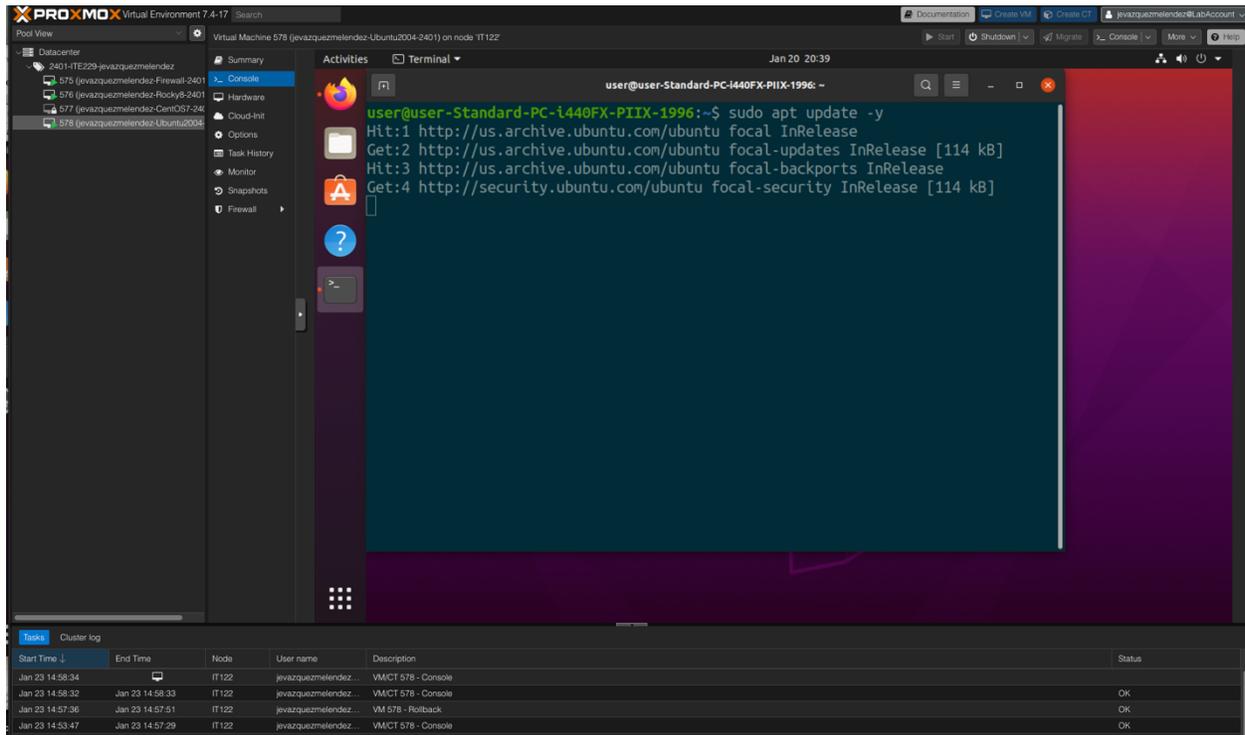
Start Time ↓	End Time	Node	User name	Description	Status
Jan 23 14:58:34		IT122	jevaqzqmellendez...	VMCT 578 - Console	
Jan 23 14:58:32	Jan 23 14:58:33	IT122	jevaqzqmellendez...	VMCT 578 - Console	OK
Jan 23 14:57:36	Jan 23 14:57:51	IT122	jevaqzqmellendez...	VM 578 - Rollback	OK
Jan 23 14:53:47	Jan 23 14:57:29	IT122	jevaqzqmellendez...	VMCT 578 - Console	OK
Jan 23 14:53:46	Jan 23 14:53:47	IT122	jevaqzqmellendez...	VM 578 - Start	OK

Update Ubuntu (refreshes the repolist)

```
$ sudo apt update -y
```

After saving the changes, you need to update the repository list by running the `sudo apt update -y`.

This command is needed so that the system can fetch the latest package information from the added repository.



The screenshot shows a Proxmox VE interface with a terminal window open. The terminal displays the command `sudo apt update -y` and its output, which includes the following lines:

```
Hit:1 http://us.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Hit:3 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease
Get:4 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
```

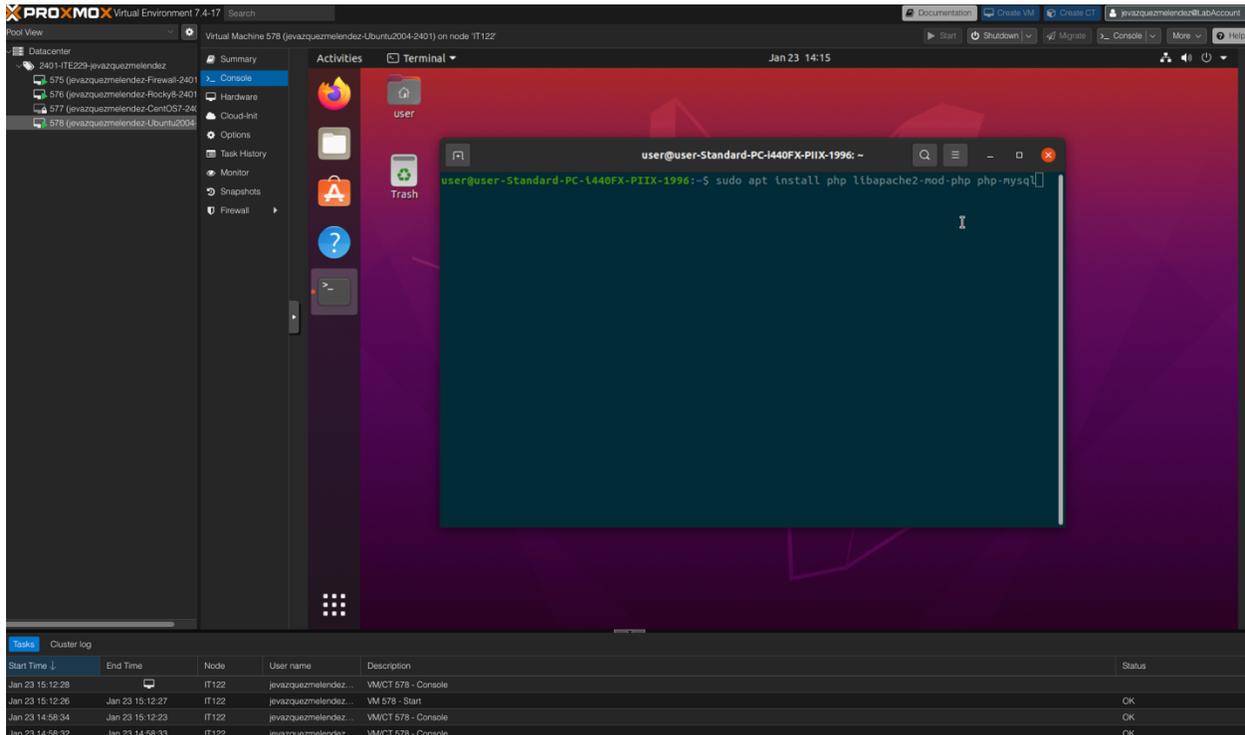
The terminal window is titled `user@user-Standard-PC-i440FX-PIIX-1996: ~` and shows the date `Jan 20 20:39`. The Proxmox VE interface includes a sidebar with a tree view of virtual machines and a task log at the bottom.

Start Time	End Time	Node	User name	Description	Status
Jan 23 14:58:34		IT122	jevasquezmelendez...	VMCT 578 - Console	
Jan 23 14:58:32	Jan 23 14:58:33	IT122	jevasquezmelendez...	VMCT 578 - Console	OK
Jan 23 14:57:36	Jan 23 14:57:51	IT122	jevasquezmelendez...	VM 578 - Rollback	OK
Jan 23 14:53:47	Jan 23 14:57:29	IT122	jevasquezmelendez...	VMCT 578 - Console	OK

Install Required PHP Libraries

```
$ sudo apt install php libapache2-mod-php php-mysql
```

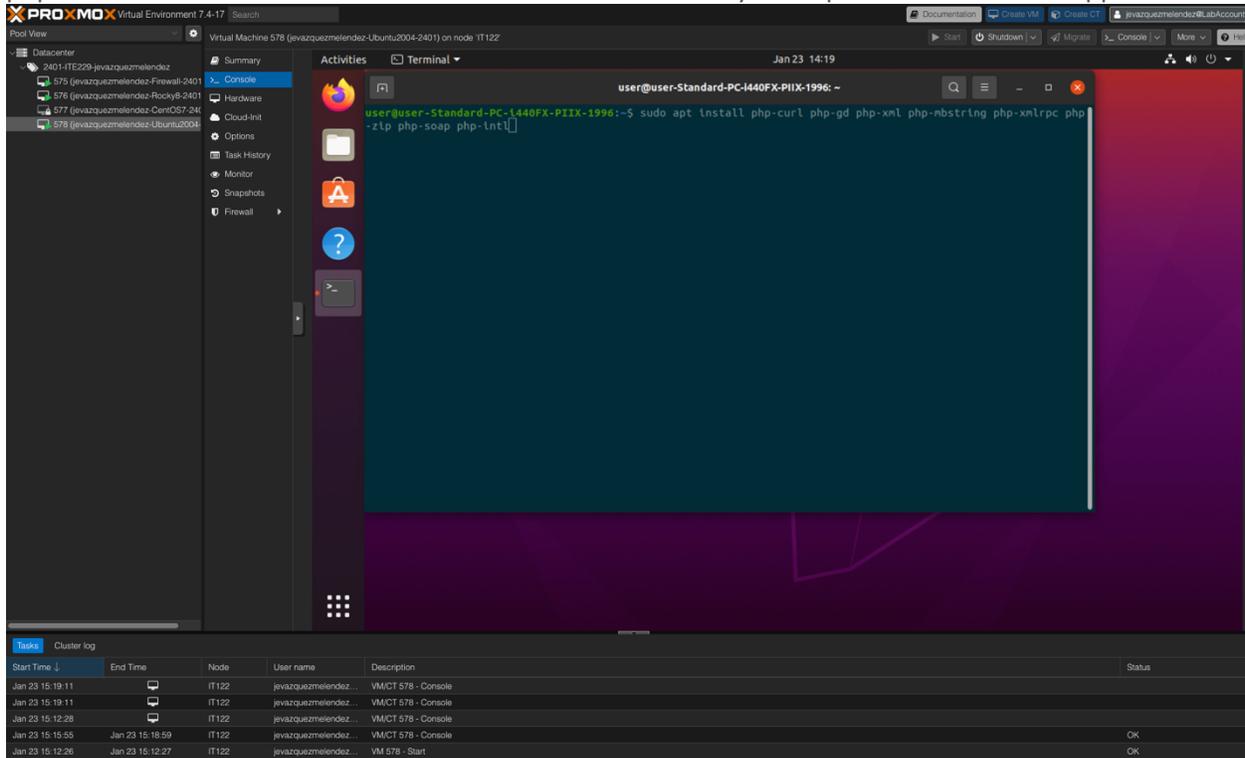
This command `sudo apt install php libapache2-mod-php php-mysql`. Is used to install PHP and its MySQL driver, as well as the Apache PHP module.



Install Required MySQL Libraries

```
$ sudo apt install php-curl php-gd php-xml php-mbstring php-xmlrpc php-zip php-soap php-intl
```

This command `sudo apt install php-curl php-gd php-xml php-mbstring php-xmlrpc php-zip php-soap php-intl`. Allow us to install additional PHP extensions that may be required for some web applications.



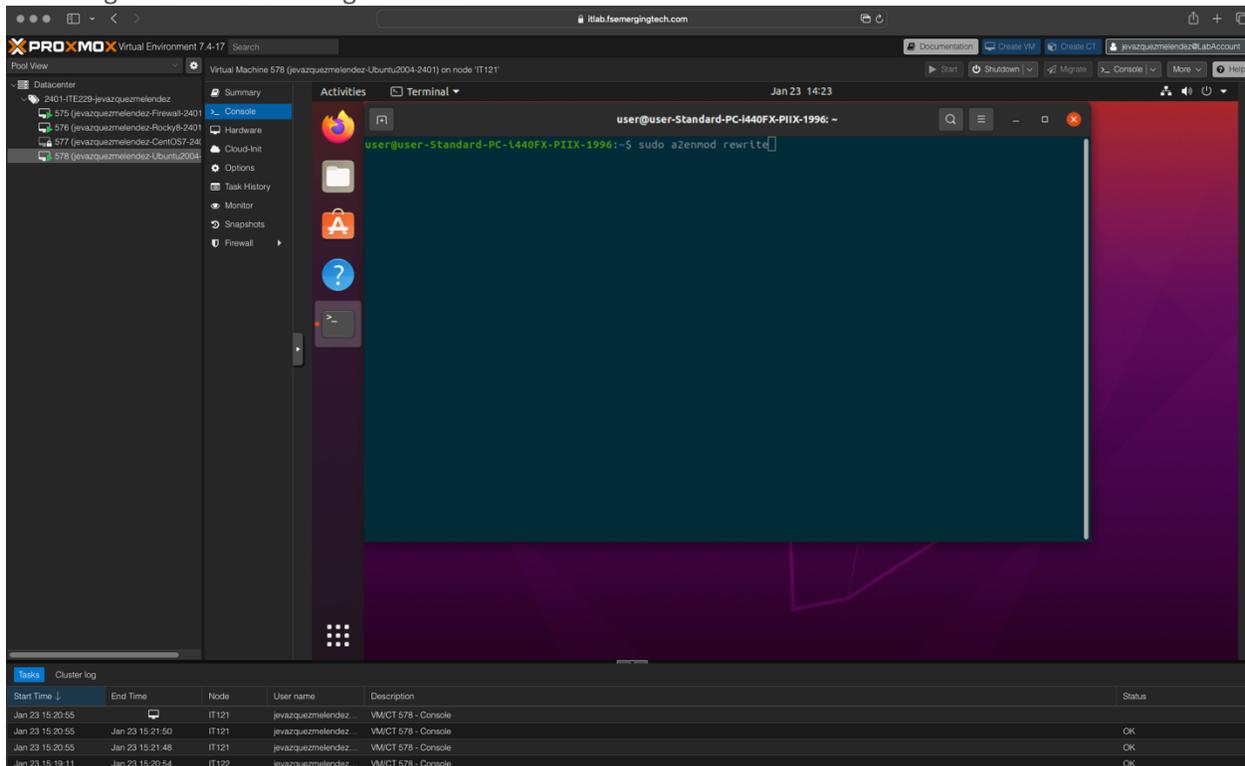
The screenshot displays the Proxmox VE interface. The main window shows a terminal session for a virtual machine named '578 (jevazquezmelendez-Ubuntu2004-2401)'. The terminal prompt is `user@user-Standard-PC-i440FX-PIIX-1996:~`. The command entered is `sudo apt install php-curl php-gd php-xml php-mbstring php-xmlrpc php-zip php-soap php-intl`. The terminal output is currently blank, indicating the command has been entered but not yet executed. The interface also shows a sidebar with navigation options like 'Summary', 'Console', 'Hardware', 'Cloud-Init', 'Options', 'Task History', 'Monitor', 'Snapshots', and 'Firewall'. At the bottom, there is a 'Tasks' table with the following data:

Start Time	End Time	Node	User name	Description	Status
Jan 23 15:19:11		IT122	jevazquezmelendez...	VM/CT 578 - Console	
Jan 23 15:19:11		IT122	jevazquezmelendez...	VM/CT 578 - Console	
Jan 23 15:12:28		IT122	jevazquezmelendez...	VM/CT 578 - Console	
Jan 23 15:15:55	Jan 23 15:18:59	IT122	jevazquezmelendez...	VM/CT 578 - Console	OK
Jan 23 15:12:26	Jan 23 15:12:27	IT122	jevazquezmelendez...	VM 578 - Start	OK

Enable URL Rewrites (clean URLs)

```
$ sudo a2enmod rewrite
```

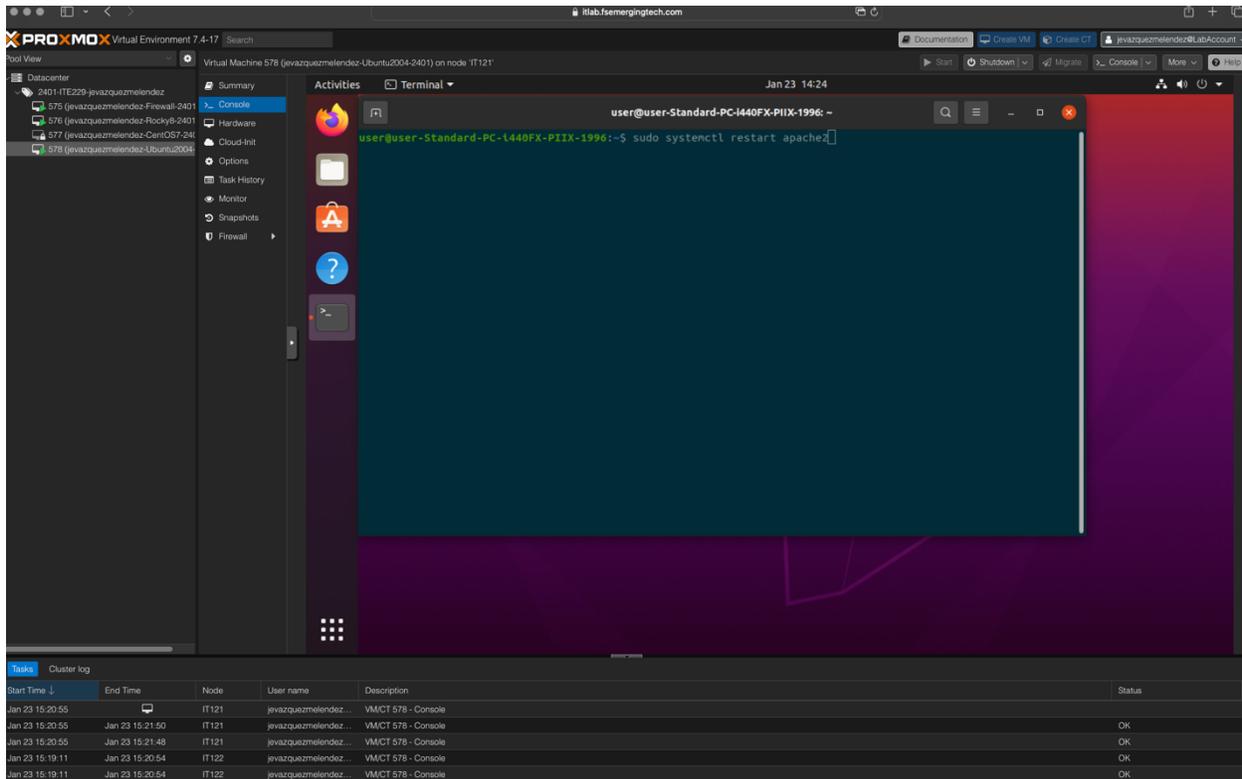
The command `sudo a2enmod rewrite`. Allow us to enable the Apache module `mod_rewrite`, which allows for URL rewriting and is commonly used in web applications to create clean URLs. This will benefit the user accessing the website making it easier.



Restart Apache Service

```
$ sudo systemctl restart apache2
```

This command, `sudo systemctl restart apache2`, is used to restart the Apache web server software. This command is usually executed when there are some changes made to the configuration files of the web server.



The screenshot displays the Proxmox Virtual Environment (VE) interface. The main window shows a terminal session for a virtual machine named '578 (jevizquzmelendez-Ubuntu2004-2401)'. The terminal prompt is 'user@user-Standard-PC-L440FX-PIIX-1996: ~'. The command 'sudo systemctl restart apache2' has been entered and is highlighted. The interface includes a sidebar with navigation options like 'Summary', 'Hardware', 'Cloud-Init', 'Options', 'Task History', 'Monitor', 'Snapshots', and 'Firewall'. At the bottom, there is a 'Tasks' table showing a cluster log.

Start Time	End Time	Node	User name	Description	Status
Jan 23 15:20:55		IT121	jevizquzmelendez...	VMCT 578 - Console	
Jan 23 15:20:55	Jan 23 15:21:50	IT121	jevizquzmelendez...	VMCT 578 - Console	OK
Jan 23 15:20:55	Jan 23 15:21:48	IT121	jevizquzmelendez...	VMCT 578 - Console	OK
Jan 23 15:19:11	Jan 23 15:20:54	IT122	jevizquzmelendez...	VMCT 578 - Console	OK
Jan 23 15:19:11	Jan 23 15:20:54	IT122	jevizquzmelendez...	VMCT 578 - Console	OK

Create a test.php Web Page

```
$ sudo nano /var/www/html/test.php
```

```
<?php phpinfo(); ?>
```

Press x and then y to save the file and exit nano.

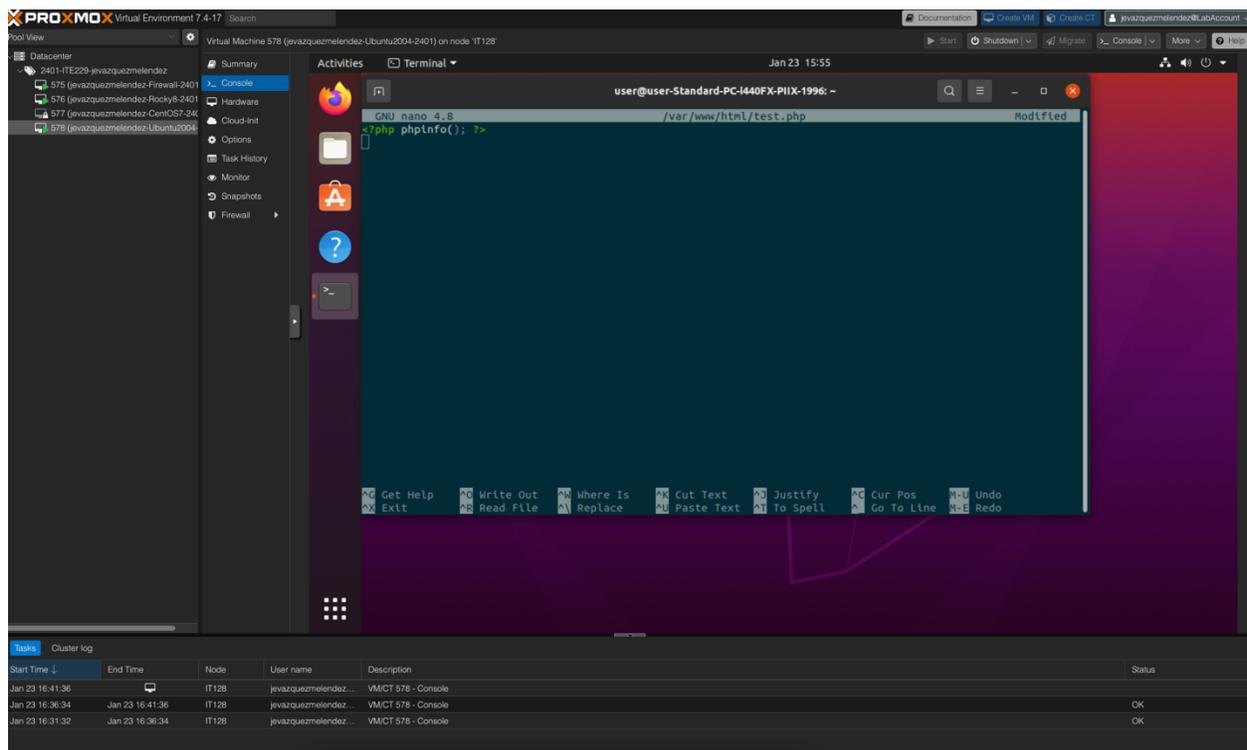
The command `sudo nano /var/www/html/test.php`, is used to open the `test.php` file in the nano text editor.

This file is located in the `/var/www/html/` directory and contains a simple PHP script that displays the PHP configuration information.

After opening the file in Nano, you need to add the code:

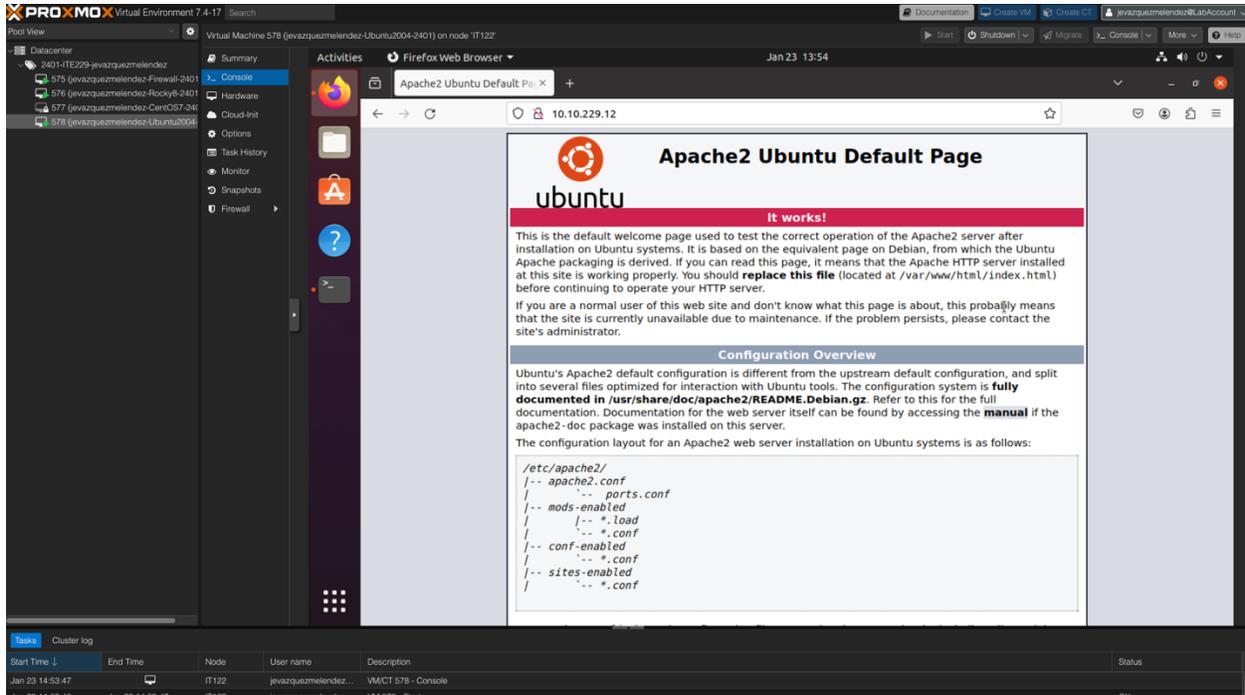
```
<?php phpinfo(); ?>
```

This code will display the PHP configuration information on the web page. To save the changes made to the file, you need to press 'x' to exit the editor and then 'y' to save the changes.



Test the test.php Web Page

Open the Firefox Web Browser. Once you have saved the changes, you can test the web page of the test.php using the IP address 10.10.229.12/test.php to connect to the server where the Apache web is running.



The screenshot shows the Proxmox VE interface. On the left, there is a sidebar with a tree view of virtual machines and a list of tasks. The main area displays a Firefox web browser window. The browser's address bar shows the URL 10.10.229.12. The page content includes the Apache2 Ubuntu logo, the text "It works!", and a "Configuration Overview" section. The configuration overview text explains that the default configuration is different from the upstream default and is split into several files. Below the text is a code block showing the directory structure for the Apache2 configuration files.

```
#!/etc/apache2/  
|-- apache2.conf  
|  
|-- ports.conf  
|-- mods-enabled  
|  
|   |-- *.load  
|   |-- *.conf  
|-- conf-enabled  
|  
|-- sites-enabled  
|  
|-- *.conf
```

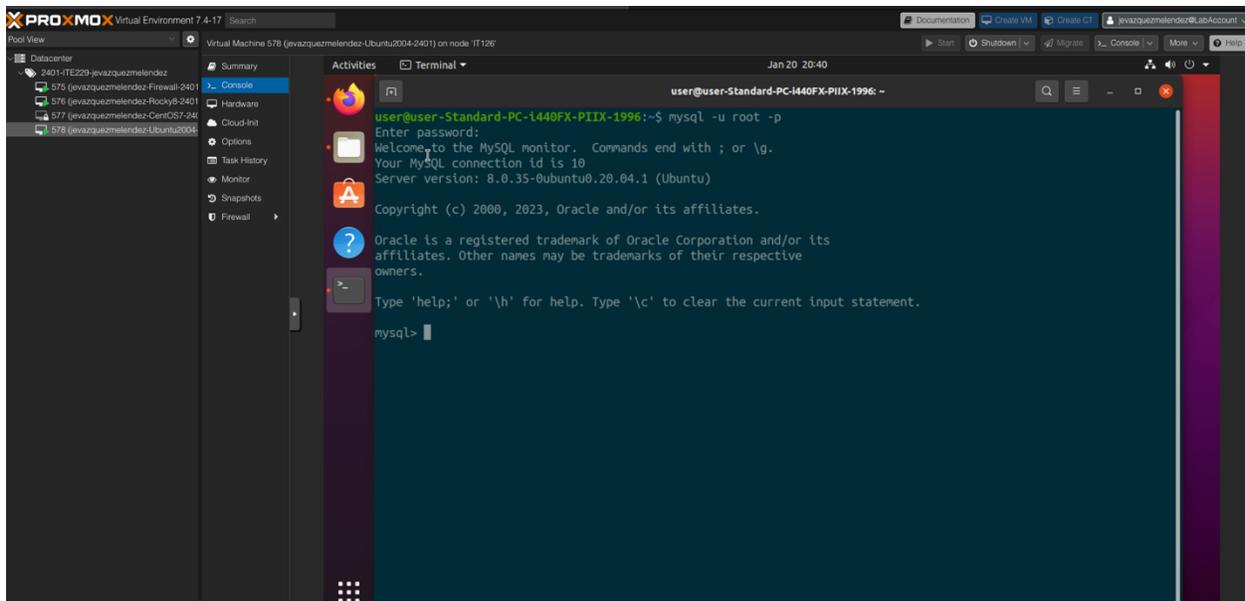
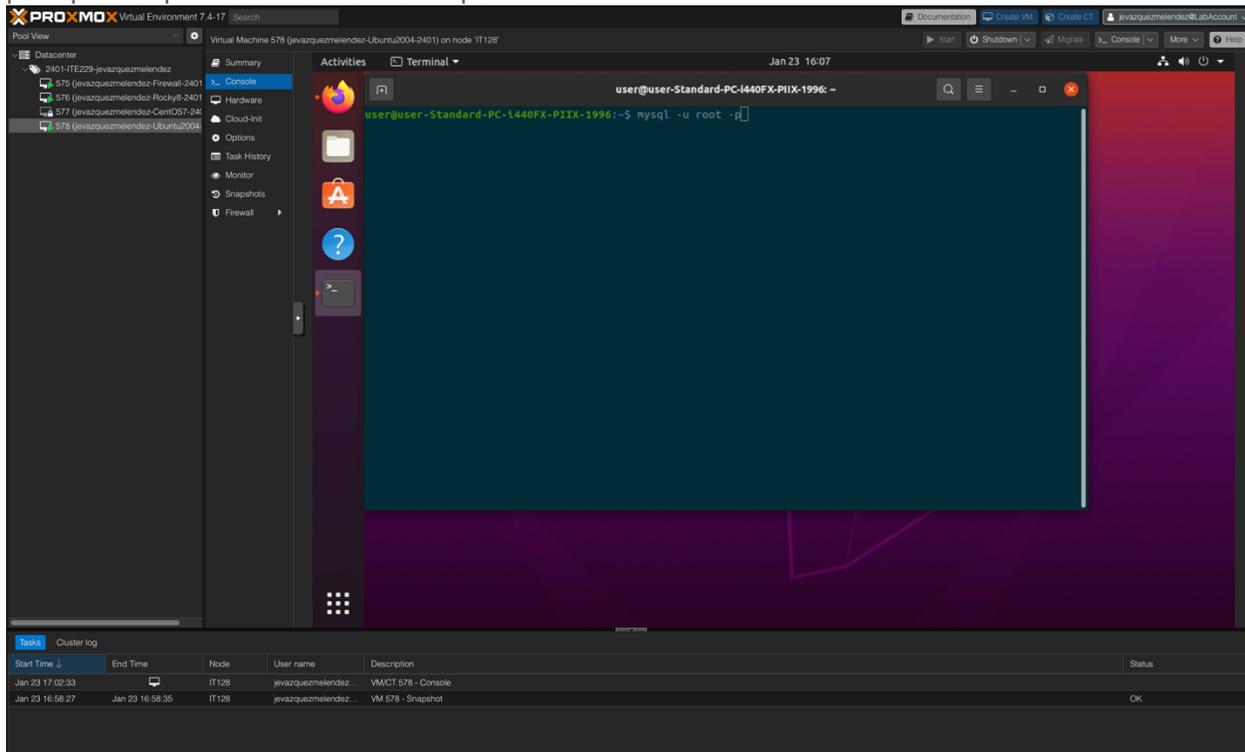
Start Time	End Time	Node	User name	Description	Status
Jan 23 14:53:47		IT122	jevaizquezmelendez...	VM/CT 578 - Console	
Jan 23 14:53:48	Jan 23 14:53:47	IT122	jevaizquezmelendez...	VM/CT 578 - Stand	OK

Database Configuration in MySQL

Log into MySQL Database

```
$ mysql -u root -p
```

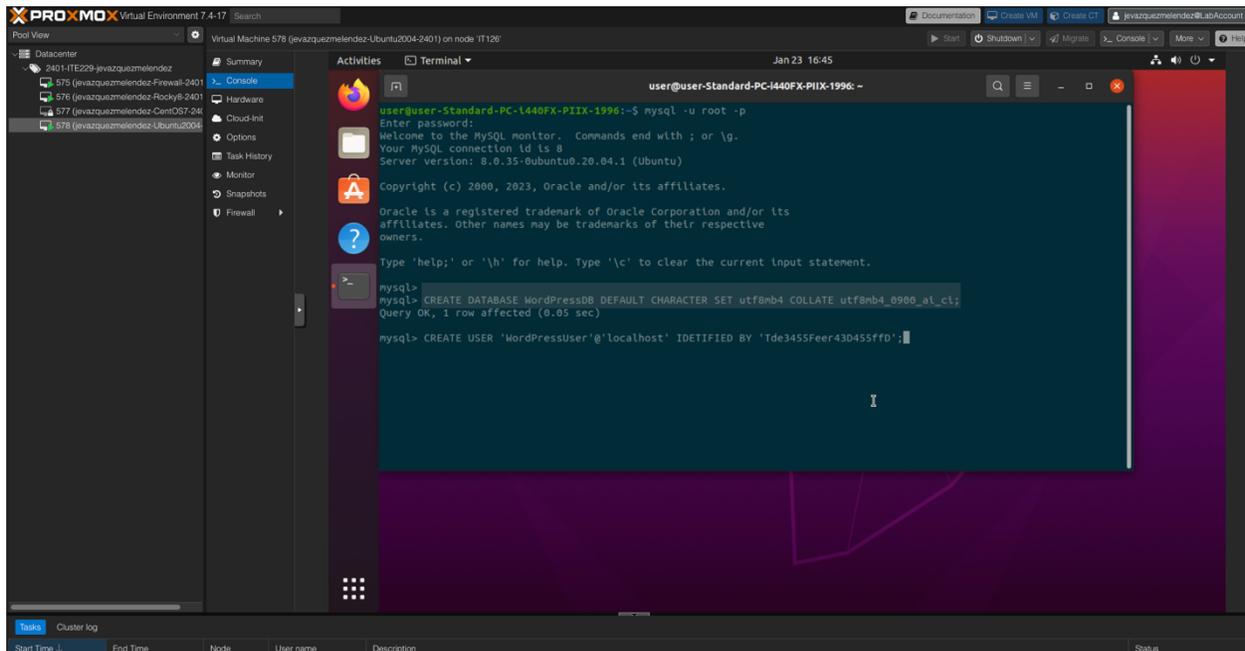
The command "mysql -u root -p" is used to log in to the MySQL database with the user "root" and it will prompt for a password. Enter the one provided.



Create WordPress Database in MySQL

```
mysql> CREATE DATABASE WordPressDB DEFAULT SET utf8mb4 COLLATE uft8mb4_0900_ai_ci;
```

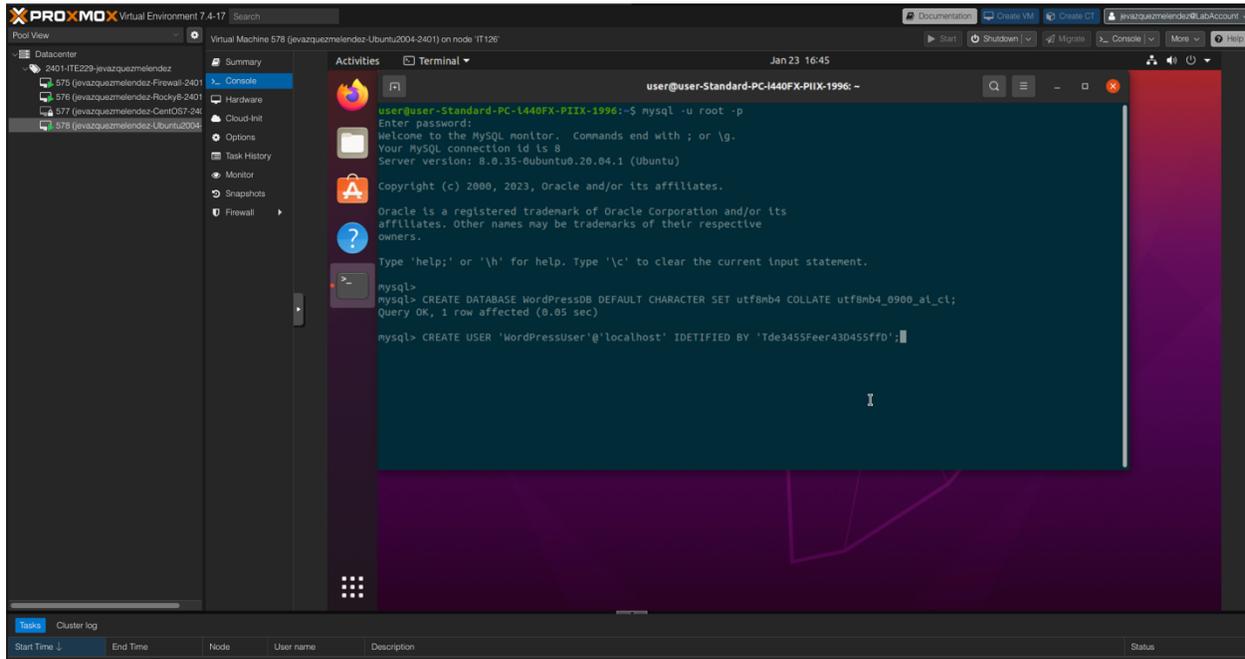
The command `CREATE DATABASE WordPressDB DEFAULT SET utf8mb4 COLLATE uft8mb4_0900_ai_ci.` is used to create a new database named "WordPressDB" with a default character set of "utf8mb4" and a specific collation of `uft8mb4_0900_ai_ci`.



Create WordPress User for MySQL Database

```
Mysql> CREATE USER 'WordPressUser'@'localhost' IDENTIFIED BY 'Tde3455Fr43D455ffd';
```

This is used to create a new user in the MySQL database system named "WordPressUser". The keyword " IDENTIFIED BY 'Tde3455Fr43D455ffd';" is used to set as a password for the user.



The screenshot shows a terminal window within a Proxmox VE environment. The terminal is running a MySQL shell as the root user. The following commands and their outputs are visible:

```
user@user-Standard-PC-1440FX-PIIX-1996:~$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.35-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
mysql> CREATE DATABASE WordPressDB DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci;
Query OK, 1 row affected (0.05 sec)

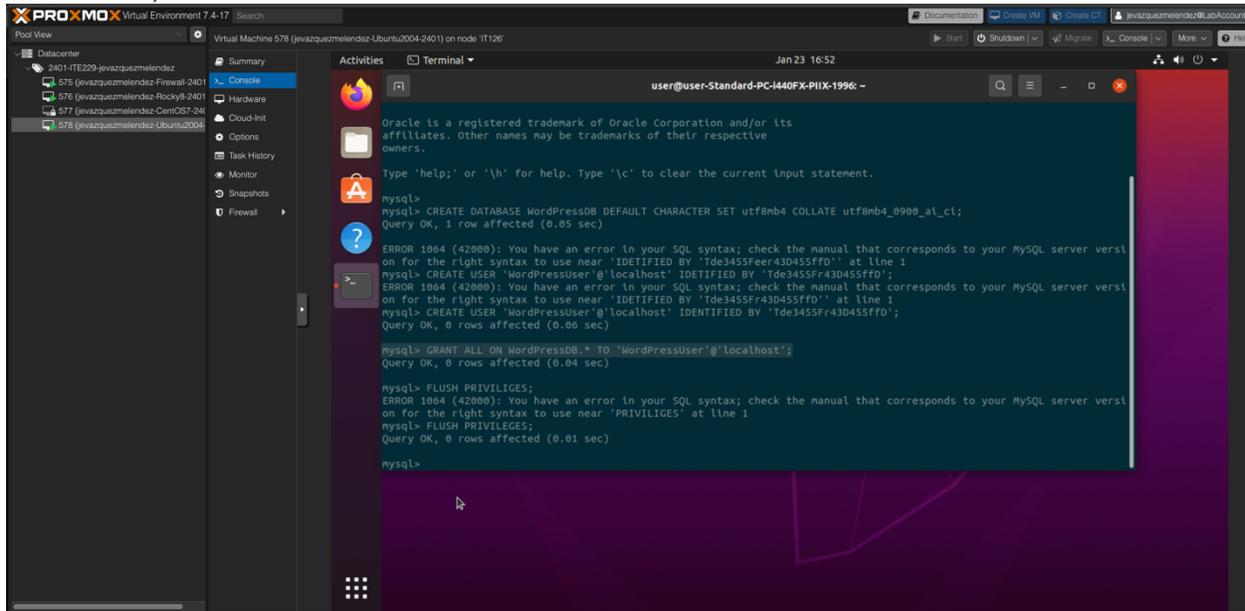
mysql> CREATE USER 'WordPressUser'@'localhost' IDENTIFIED BY 'Tde3455Fr43D455ffd';
```

The terminal window also shows the Proxmox VE interface on the left, including a sidebar with navigation options like 'Datacenter', 'Summary', 'Console', 'Hardware', 'Cloud-Init', 'Options', 'Task History', 'Monitor', 'Snapshots', and 'Firewall'. The top of the terminal window displays the date and time: 'Jan 23 16:45'.

Grant Privileges to this New WordPress User

```
Mysql> GRANT ALL ON WordPressDB.* TO 'WordPressUser'@'localhost';
```

The command `mysql> GRANT ALL ON WordPressDB.* TO 'WordPressUser'@'localhost'`, is used to grant all privileges to a user named "WordPressUser" on the newly created "WordPressDB" database, and this user can only access the database from the localhost.



The screenshot shows a terminal window within a Proxmox virtual environment. The terminal displays the following MySQL commands and their outputs:

```
mysql> CREATE DATABASE WordPressDB DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci;
Query OK, 1 row affected (0.05 sec)

mysql> CREATE USER 'WordPressUser'@'localhost' IDENTIFIED BY 'Tde3455Fr43D455FFD';
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'IDENTIFIED BY 'Tde3455Fr43D455FFD'' at line 1

mysql> CREATE USER 'WordPressUser'@'localhost' IDENTIFIED BY 'Tde3455Fr43D455FFD';
Query OK, 0 rows affected (0.06 sec)

mysql> GRANT ALL ON WordPressDB.* TO 'WordPressUser'@'localhost';
Query OK, 0 rows affected (0.04 sec)

mysql> FLUSH PRIVILEGES;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'PRIVILEGES' at line 1

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.01 sec)

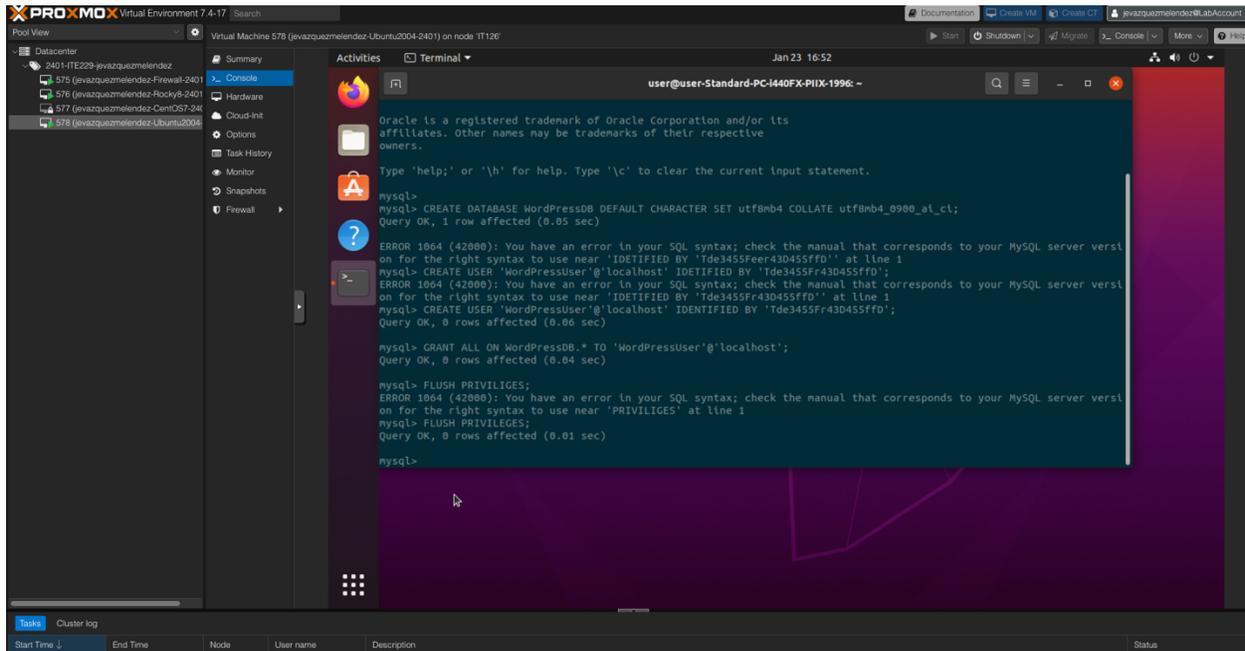
mysql>
```

Flush Privileges

```
mysql> FLUSH PRIVILEGES;
```

The command `mysql> FLUSH PRIVILEGES;`

It is used to apply the changes made in the previous step and reload the privileges table in the database.



The screenshot shows a terminal window within a Proxmox Virtual Environment. The terminal displays the following MySQL commands and their outputs:

```
mysql> CREATE DATABASE WordPressDB DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci;
Query OK, 1 row affected (0.05 sec)

mysql> CREATE USER 'WordPressUser'@'localhost' IDENTIFIED BY 'Tde3455Fr43D455ffD';
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'IDENTIFIED BY 'Tde3455Fr43D455ffD'' at line 1

mysql> CREATE USER 'WordPressUser'@'localhost' IDENTIFIED BY 'Tde3455Fr43D455ffD';
Query OK, 0 rows affected (0.06 sec)

mysql> GRANT ALL ON WordPressDB.* TO 'WordPressUser'@'localhost';
Query OK, 0 rows affected (0.04 sec)

mysql> FLUSH PRIVILEGES;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'PRIVILEGES' at line 1

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.01 sec)

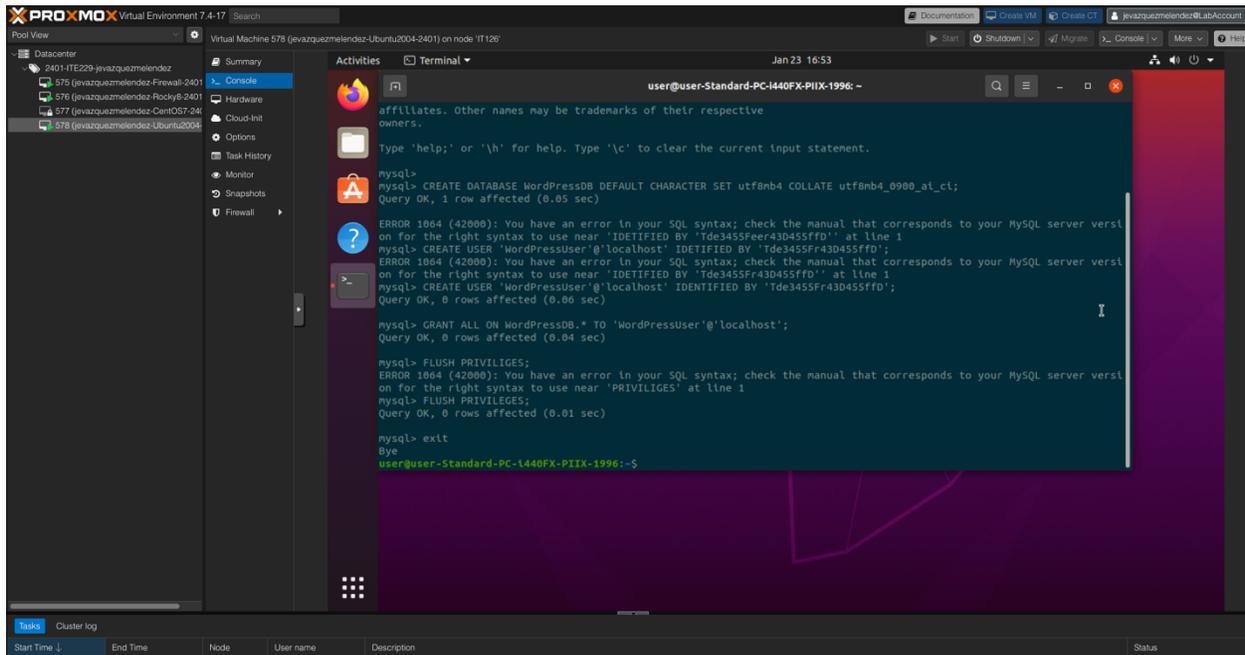
mysql>
```

The terminal window also shows a sidebar with various system icons and a top navigation bar with options like 'Start', 'Shutdown', 'Migrate', 'Console', and 'Help'.

Quit MySQL

```
mysql> exit
```

The exit command is used to quit the MySQL command-line interface. When you type this command and press Enter, the MySQL shell will close, and you will return to the ubuntu shell.



The screenshot shows a terminal window within a Proxmox Virtual Environment. The terminal displays the following sequence of commands and outputs:

```
user@user-Standard-PC-1440FX-PIIX-1996: ~$ mysql
mysql> CREATE DATABASE WordPressDB DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci;
Query OK, 1 row affected (0.05 sec)

mysql> CREATE USER 'WordPressUser'@'localhost' IDENTIFIED BY 'Tde3455Fr43D455Ffd';
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'IDENTIFIED BY 'Tde3455Fr43D455Ffd'' at line 1
mysql> CREATE USER 'WordPressUser'@'localhost' IDENTIFIED BY 'Tde3455Fr43D455Ffd';
Query OK, 0 rows affected (0.06 sec)

mysql> GRANT ALL ON WordPressDB.* TO 'WordPressUser'@'localhost';
Query OK, 0 rows affected (0.04 sec)

mysql> FLUSH PRIVILEGES;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'PRIVILEGES' at line 1
mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.01 sec)

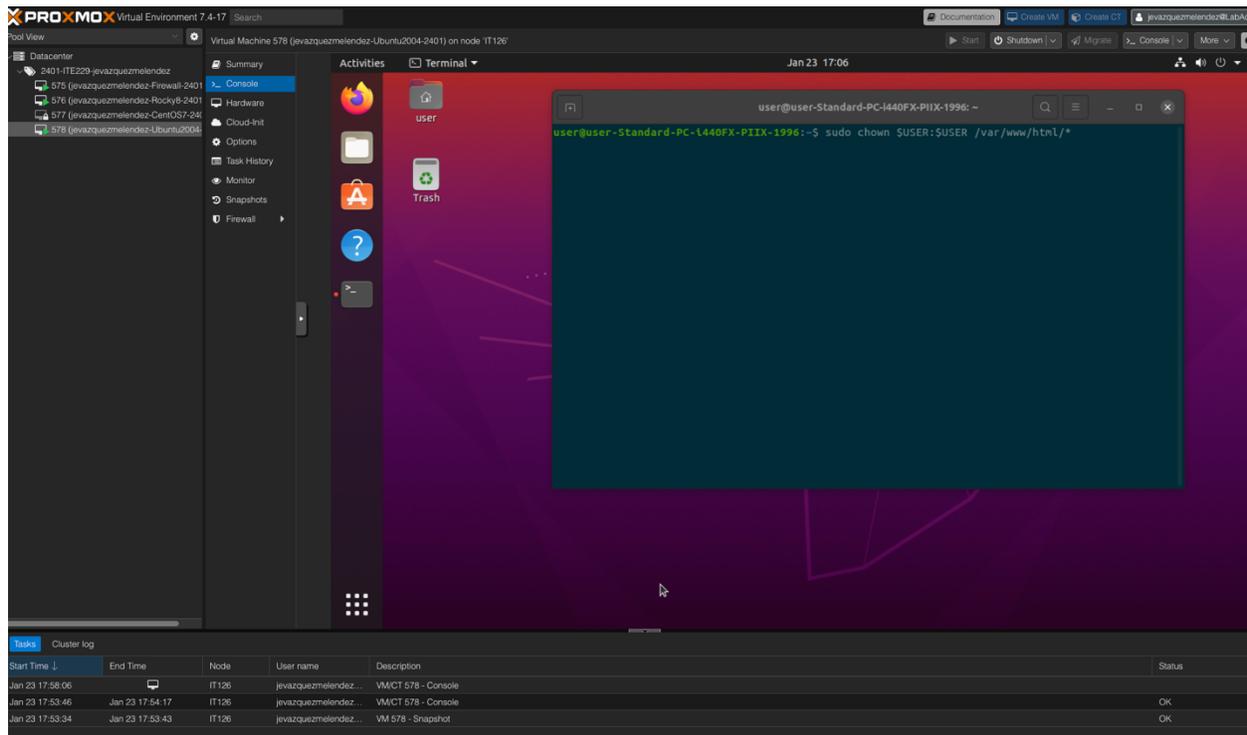
mysql> exit
Bye
user@user-Standard-PC-1440FX-PIIX-1996:~$
```

Install WordPress

Grant Permission to html Directory to WordPress User

```
$ sudo chown $USER:$USER /var/www/html/*
```

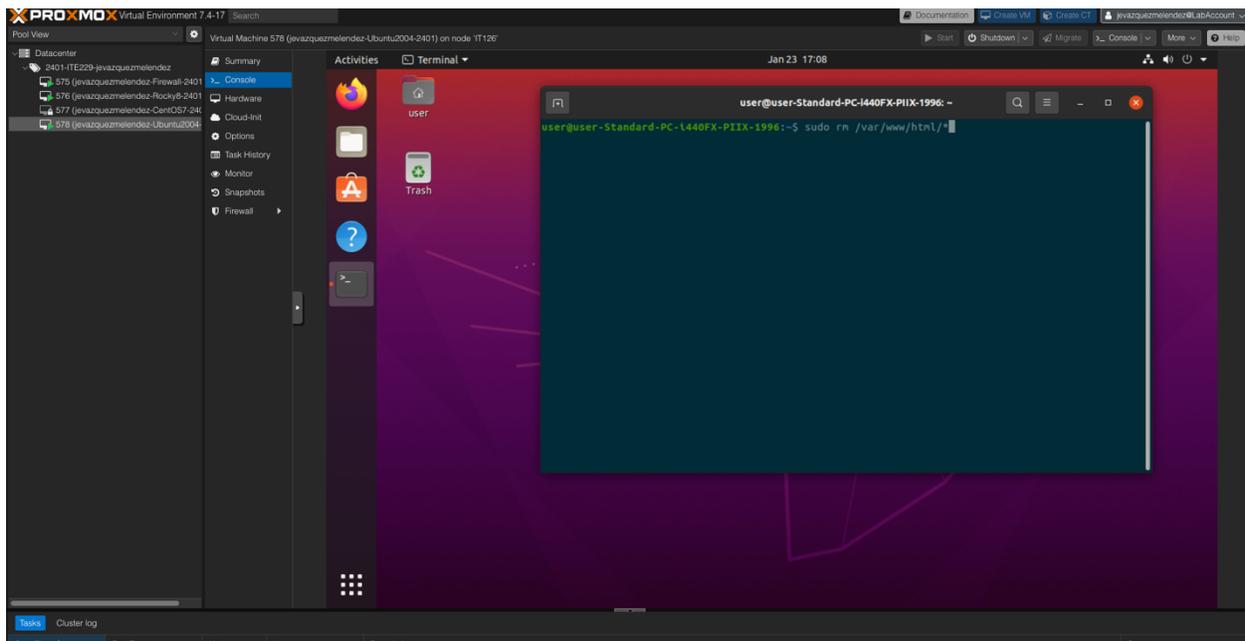
This command is going to allow us to change the ownership of all files and directories inside the /var/www/html folder to the current user.



Delete Files from html Directory

```
$ sudo rm /var/www/html/*
```

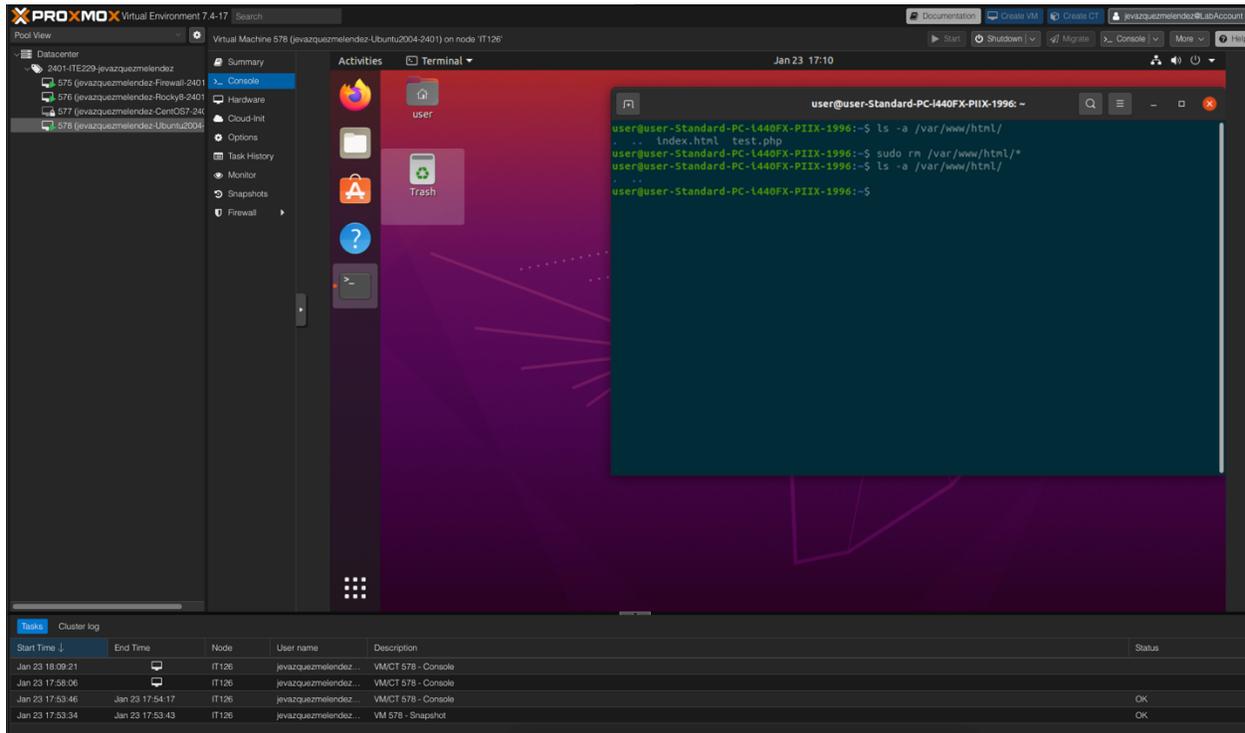
To Remove files in the html directory we are going to use `sudo rm /var/www/html/*`. This command will delete all files and directories inside the `/var/www/html` folder.



Verify html Directory is Empty

```
$ ls -a /var/www/html/
```

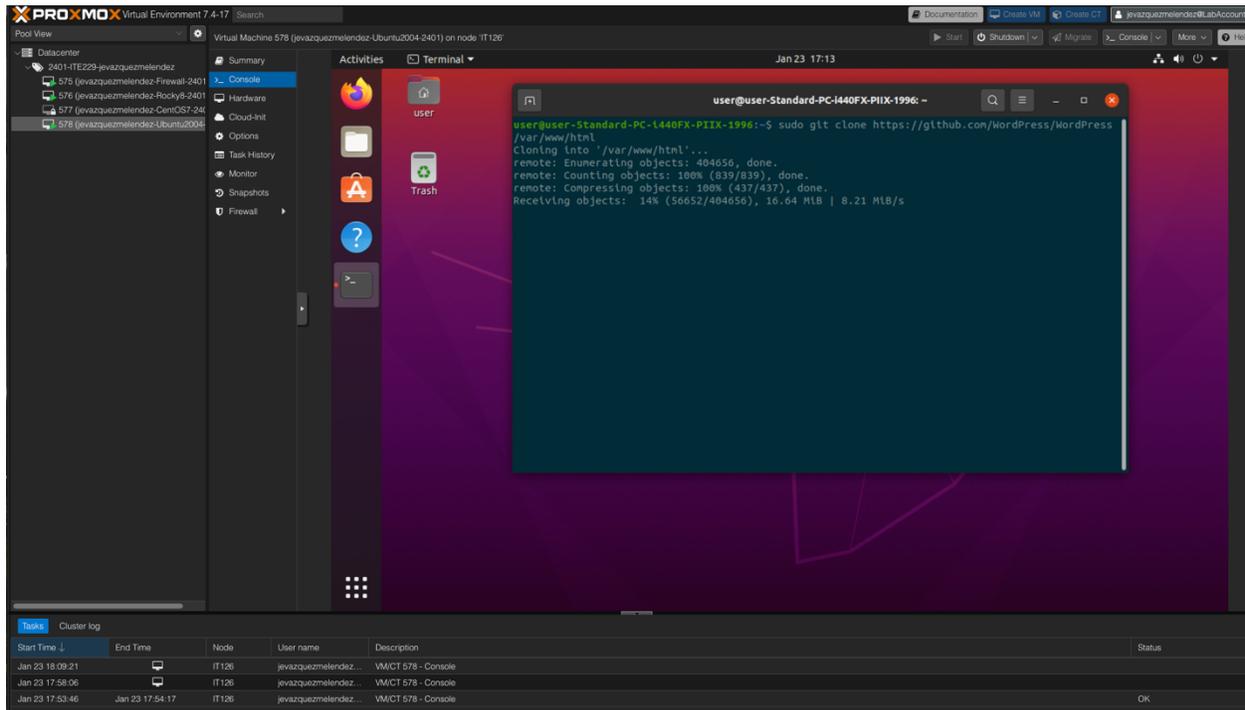
This command uses `ls -a /var/www/html/` to lists all files and directories inside the `/var/www/html` folder, including the hidden ones.



Clone WordPress to html Directory

```
$ sudo git clone https://github.com/WordPress/WordPress/ /var/www/html
```

This is the command that we are going to use to clone a Git repository. We are cloning it from a specified URL and saves it in the /var/www/html folder.



The screenshot shows a Proxmox Virtual Environment interface. A terminal window is open, displaying the execution of the command to clone the WordPress repository. The output shows the cloning process is complete, with 14k objects received at a rate of 10.64 MiB/s.

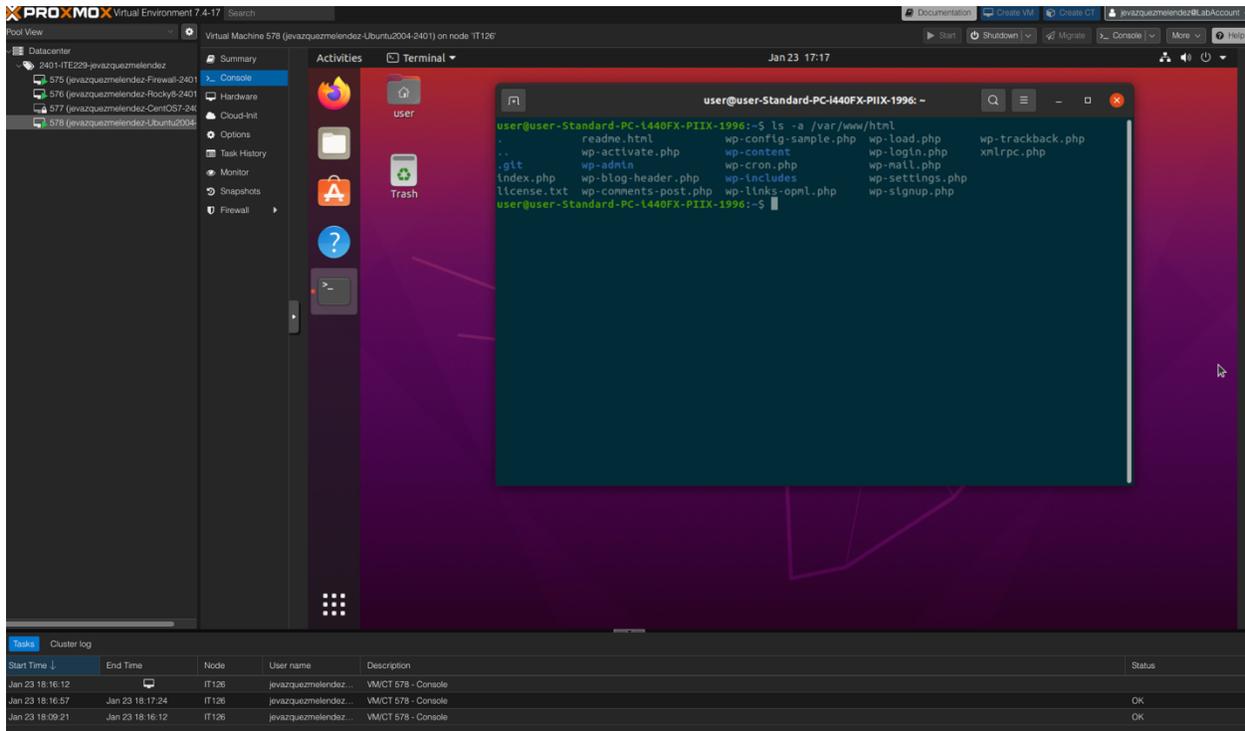
```
user@user-Standard-PC-I440FX-PIIX-1996:~$ sudo git clone https://github.com/WordPress/WordPress /var/www/html
Cloning into '/var/www/html'...
remote: Enumerating objects: 404656, done.
remote: Counting objects: 100% (839/839), done.
remote: Compressing objects: 100% (437/437), done.
Receiving objects: 14k (56652/404656), 10.64 MiB | 8.21 MiB/s
```

Start Time	End Time	Node	User name	Description	Status
Jan 23 18:09:21		IT126	jevizquezmelendez...	VMCT 578 - Console	
Jan 23 17:58:06		IT126	jevizquezmelendez...	VMCT 578 - Console	
Jan 23 17:53:46	Jan 23 17:54:17	IT126	jevizquezmelendez...	VMCT 578 - Console	OK

Verify html Directory Contains WordPress Files

```
$ ls -a /var/www/html/
```

This command uses `ls -a /var/www/html` to lists all files and directories inside the `/var/www/html` folder, including the hidden ones.



The screenshot shows a Proxmox VE interface with a terminal window open. The terminal displays the output of the command `ls -a /var/www/html`, listing various files and directories including hidden ones like `readme.html`, `wp-config-sample.php`, `wp-load.php`, `wp-trackback.php`, `wp-activate.php`, `wp-content`, `wp-login.php`, `xmlrpc.php`, `wp-admin`, `wp-cron.php`, `wp-mail.php`, `index.php`, `wp-blog-header.php`, `wp-includes`, `wp-settings.php`, `license.txt`, `wp-comments-post.php`, `wp-links-opml.php`, and `wp-signup.php`.

Start Time	End Time	Node	User name	Description	Status
Jan 23 18:16:12		IT126	jevasquezmelendez...	VM/CT 578 - Console	
Jan 23 18:16:57	Jan 23 18:17:24	IT126	jevasquezmelendez...	VM/CT 578 - Console	OK
Jan 23 18:09:21	Jan 23 18:16:12	IT126	jevasquezmelendez...	VM/CT 578 - Console	OK

Verify Permissions on html Directory

```
$ sudo ls -ls /var/www/html
```

Typing the following command `sudo ls -ls /var/www/html`. A list of all files and directories inside the `/var/www/html` folder with detailed information about their size and permissions.

```
user@user-Standard-PC-L440FX-PIIX-1996:~$ sudo ls -ls /var/www/html
total 228
 4 -rwxr--r-- 1 root root 405 Jan 23 17:15 index.php
20 -rw-r--r-- 1 root root 19915 Jan 23 17:15 license.txt
 8 -rwxr--r-- 1 root root 7401 Jan 23 17:15 readme.html
 8 -rwxr--r-- 1 root root 7222 Jan 23 17:15 wp-activate.php
 4 drwxr-xr-x 9 root root 4096 Jan 23 17:15 wp-admin
 4 -rwxr--r-- 1 root root 351 Jan 23 17:15 wp-blog-header.php
 4 -rwxr--r-- 1 root root 2323 Jan 23 17:15 wp-comments-post.php
 4 -rwxr--r-- 1 root root 3012 Jan 23 17:15 wp-config-sample.php
 4 drwxr-xr-x 4 root root 4096 Jan 23 17:15 wp-content
 8 -rwxr--r-- 1 root root 5638 Jan 23 17:15 wp-cron.php
12 drwxr-xr-x 28 root root 12288 Jan 23 17:15 wp-includes
 4 -rwxr--r-- 1 root root 2502 Jan 23 17:15 wp-links-opml.php
 4 -rwxr--r-- 1 root root 3927 Jan 23 17:15 wp-load.php
52 -rwxr--r-- 1 root root 50917 Jan 23 17:15 wp-login.php
12 -rwxr--r-- 1 root root 8525 Jan 23 17:15 wp-mail.php
28 -rwxr--r-- 1 root root 26984 Jan 23 17:15 wp-settings.php
36 -rwxr--r-- 1 root root 34385 Jan 23 17:15 wp-signup.php
 8 -rwxr--r-- 1 root root 4885 Jan 23 17:15 wp-trackback.php
 4 -rwxr--r-- 1 root root 3154 Jan 23 17:15 xmlrpc.php
user@user-Standard-PC-L440FX-PIIX-1996:~$
```

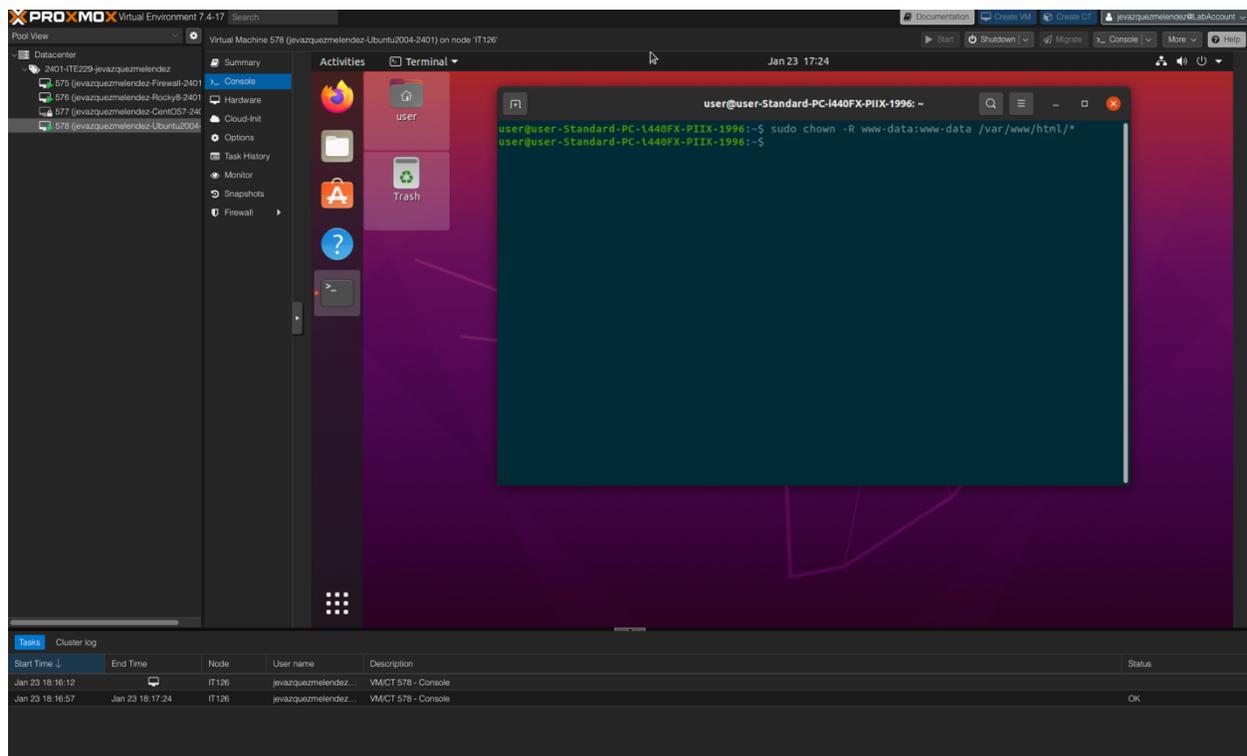
Start Time	End Time	Node	User name	Description	Status
Jan 23 18:16:12		IT126	jevaquezmelendez...	VM/CT 578 - Console	
Jan 23 18:16:57	Jan 23 18:17:24	IT126	jevaquezmelendez...	VM/CT 578 - Console	OK
Jan 23 18:09:21	Jan 23 18:16:12	IT126	jevaquezmelendez...	VM/CT 578 - Console	OK

Edit Ownership

Edit Ownership of Contents of html Directory

```
$ sudo chown -R www-data:www-data /var/www/html/*
```

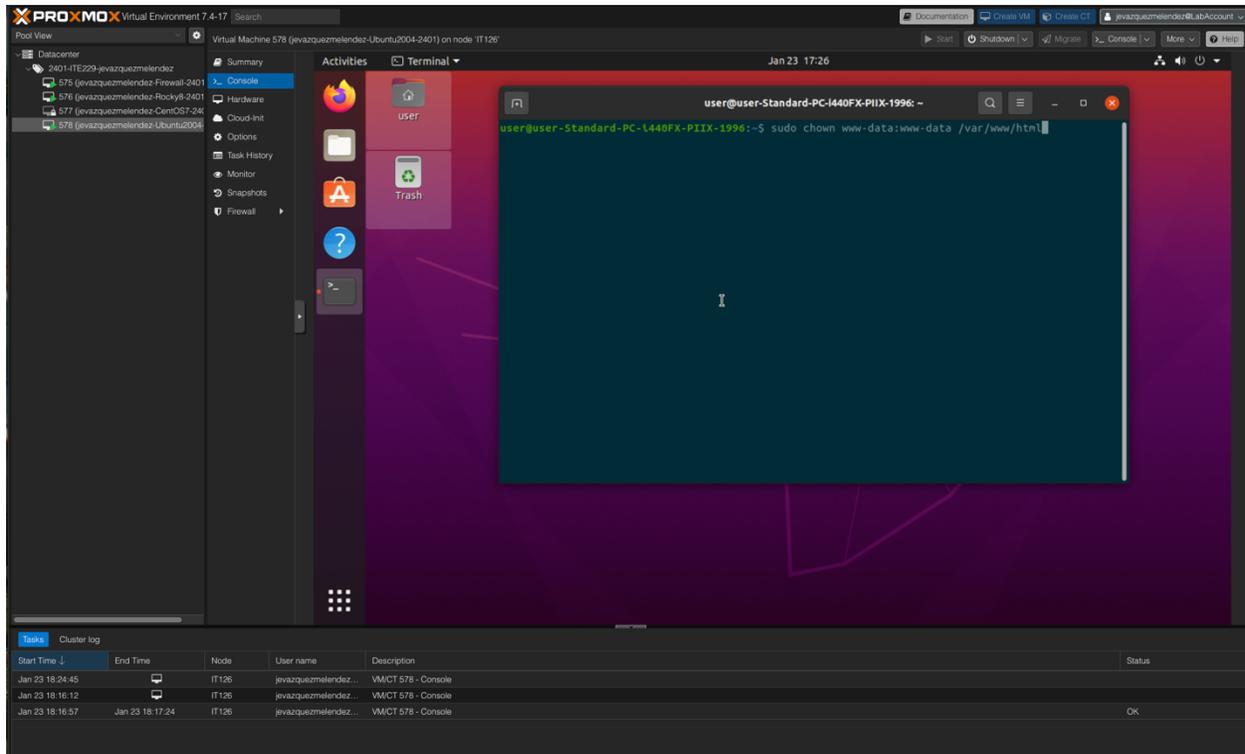
The following command using sudo privileges `sudo chown -R www-data:www-data /var/www/html/` will make changes to the ownership of all files and directories inside the `/var/www/html` folder to the `www-data` user and group.



Edit Ownership of the html Directory Itself

```
$ sudo chown www-data:www-data /var/www/html
```

To edit ownership of the html Directory we use the following command `sudo chown www-data:www-data /var/www/html`. This will make changes in the ownership of `/var/www/html` folder and assign it to the `www-data` user and group.



\$ sudo ls -ls /var/www/html

To verify the changes are made correctly.

The screenshot shows a Proxmox VE interface with a terminal window open. The terminal displays the output of the command 'sudo ls -ls /var/www/html'. The output lists various files and directories with their permissions, owners, sizes, and timestamps. The files listed include index.php, license.txt, readme.html, wp-activate.php, wp-admin, wp-blog-header.php, wp-comments-post.php, wp-config-sample.php, wp-content, wp-cron.php, wp-includes, wp-links-opml.php, wp-load.php, wp-login.php, wp-mail.php, wp-settings.php, wp-signup.php, wp-trackback.php, and xmlrpc.php.

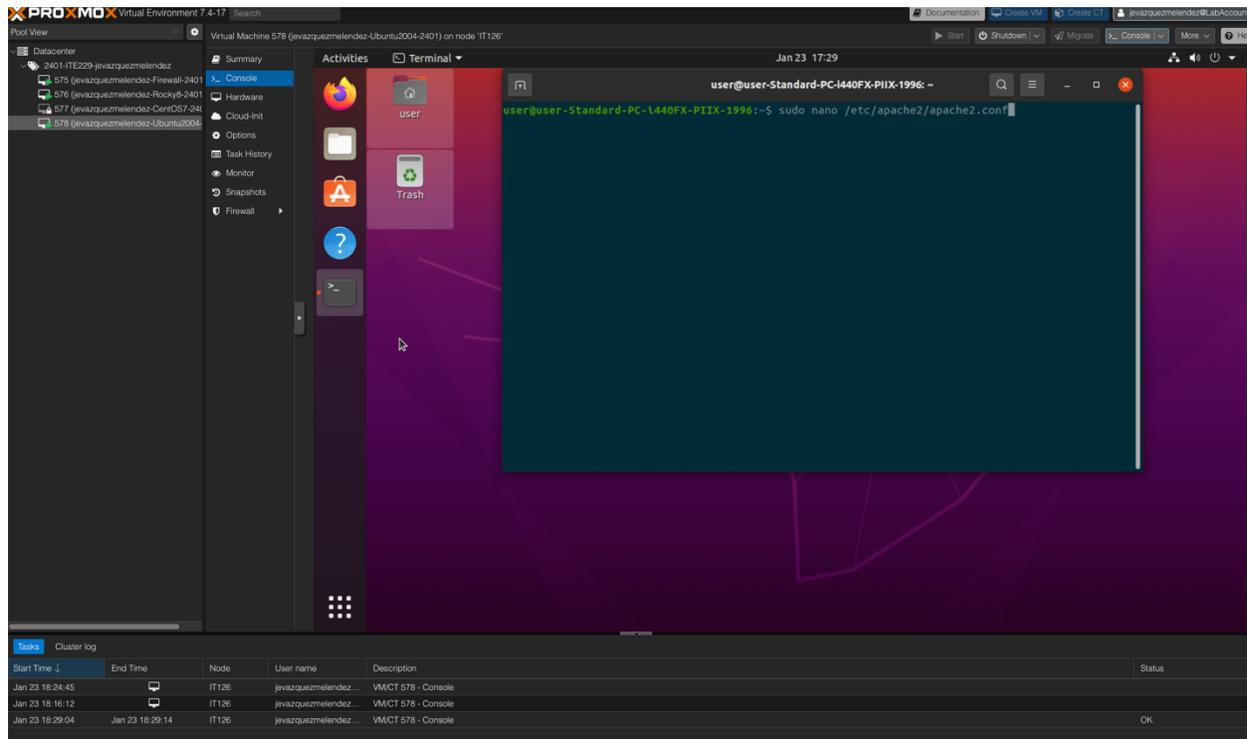
```
user@user-Standard-PC-L440FX-PIIX-1996:~$ sudo chown -R www-data:www-data /var/www/html/*
user@user-Standard-PC-L440FX-PIIX-1996:~$ sudo ls -ls /var/www/html
total 228
 4 -rw-r--r-- 1 www-data www-data 405 Jan 23 17:15 index.php
20 -rw-r--r-- 1 www-data www-data 19915 Jan 23 17:15 license.txt
 8 -rw-r--r-- 1 www-data www-data 7401 Jan 23 17:15 readme.html
 8 -rw-r--r-- 1 www-data www-data 7222 Jan 23 17:15 wp-activate.php
 4 drwxr-xr-x 9 www-data www-data 4096 Jan 23 17:15 wp-admin
 4 -rw-r--r-- 1 www-data www-data 351 Jan 23 17:15 wp-blog-header.php
 4 -rw-r--r-- 1 www-data www-data 2323 Jan 23 17:15 wp-comments-post.php
 4 -rw-r--r-- 1 www-data www-data 3012 Jan 23 17:15 wp-config-sample.php
 4 drwxr-xr-x 4 www-data www-data 4096 Jan 23 17:15 wp-content
 8 -rw-r--r-- 1 www-data www-data 5038 Jan 23 17:15 wp-cron.php
12 drwxr-xr-x 28 www-data www-data 12288 Jan 23 17:15 wp-includes
 4 -rw-r--r-- 1 www-data www-data 2502 Jan 23 17:15 wp-links-opml.php
 4 -rw-r--r-- 1 www-data www-data 3927 Jan 23 17:15 wp-load.php
52 -rw-r--r-- 1 www-data www-data 50917 Jan 23 17:15 wp-login.php
12 -rw-r--r-- 1 www-data www-data 8525 Jan 23 17:15 wp-mail.php
28 -rw-r--r-- 1 www-data www-data 26984 Jan 23 17:15 wp-settings.php
36 -rw-r--r-- 1 www-data www-data 34385 Jan 23 17:15 wp-signup.php
 8 -rw-r--r-- 1 www-data www-data 4885 Jan 23 17:15 wp-trackback.php
 4 -rw-r--r-- 1 www-data www-data 3154 Jan 23 17:15 xmlrpc.php
user@user-Standard-PC-L440FX-PIIX-1996:~$
```

Start Time	End Time	Node	User name	Description	Status
Jan 23 18:24:45		IT126	jevazquzmelendez...	VMCT 578 - Console	
Jan 23 18:16:12		IT126	jevazquzmelendez...	VMCT 578 - Console	
Jan 23 18:16:57	Jan 23 18:17:24	IT126	jevazquzmelendez...	VMCT 578 - Console	OK

Edit the apache2.conf File

```
$ sudo nano /etc/apache2/apache2.conf
```

Using this command to edit our apache2.conf file using nano our text editor. In this location is where the Apache configuration file is on your system. When you run this command, it will open the configuration file in the nano text editor, allowing you to make changes to the file as needed.

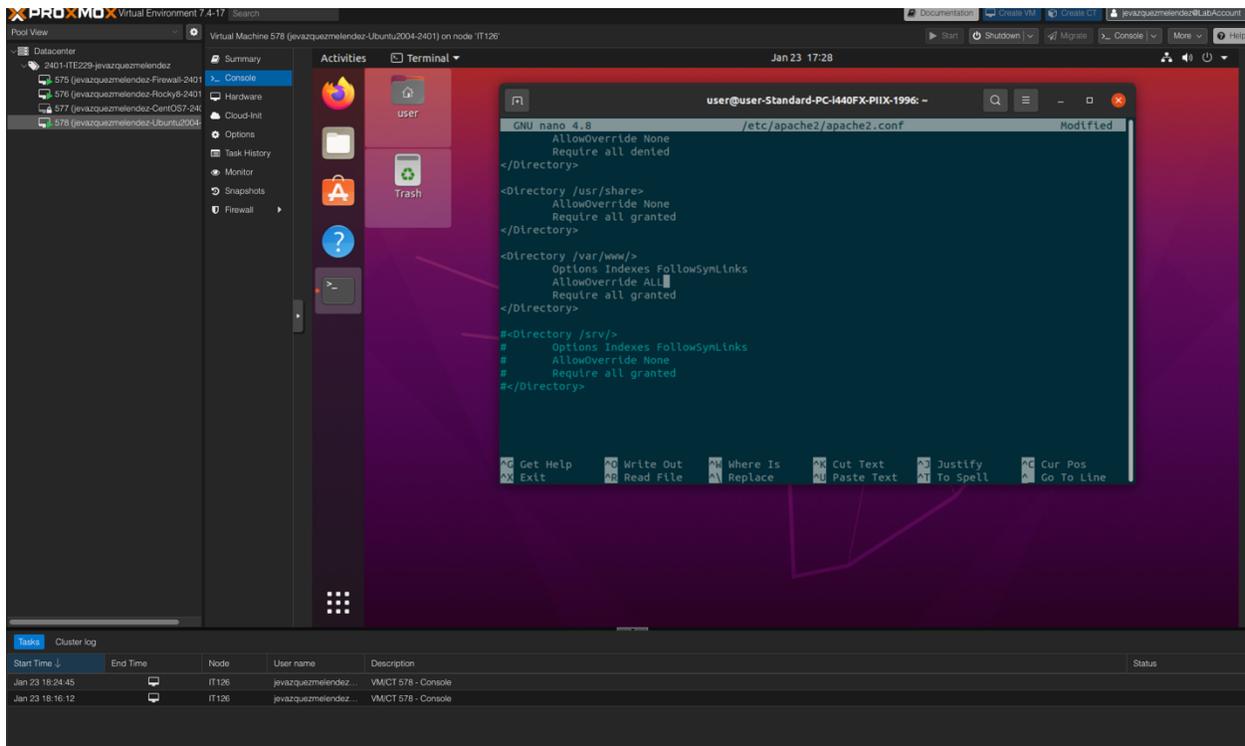


Override All Default Apache Directives

Here we are going to change the default setting to:

AllowOverride "ALL"

This will add an extra layer of security.



The screenshot shows a Proxmox VE interface with a terminal window open. The terminal is editing the file `/etc/apache2/apache2.conf` using nano 4.8. The configuration for the `<Directory /var/www/>` section is as follows:

```
<Directory /var/www/>
  Options Indexes FollowSymLinks
  AllowOverride ALL
  Require all granted
</Directory>
```

Other sections visible in the terminal include:

```
AllowOverride None
Require all denied
</Directory>

<Directory /usr/share>
  AllowOverride None
  Require all granted
</Directory>

#<Directory /srv/>
#   Options Indexes FollowSymLinks
#   AllowOverride None
#   Require all granted
#</Directory>
```

The terminal also shows a table of cluster logs at the bottom:

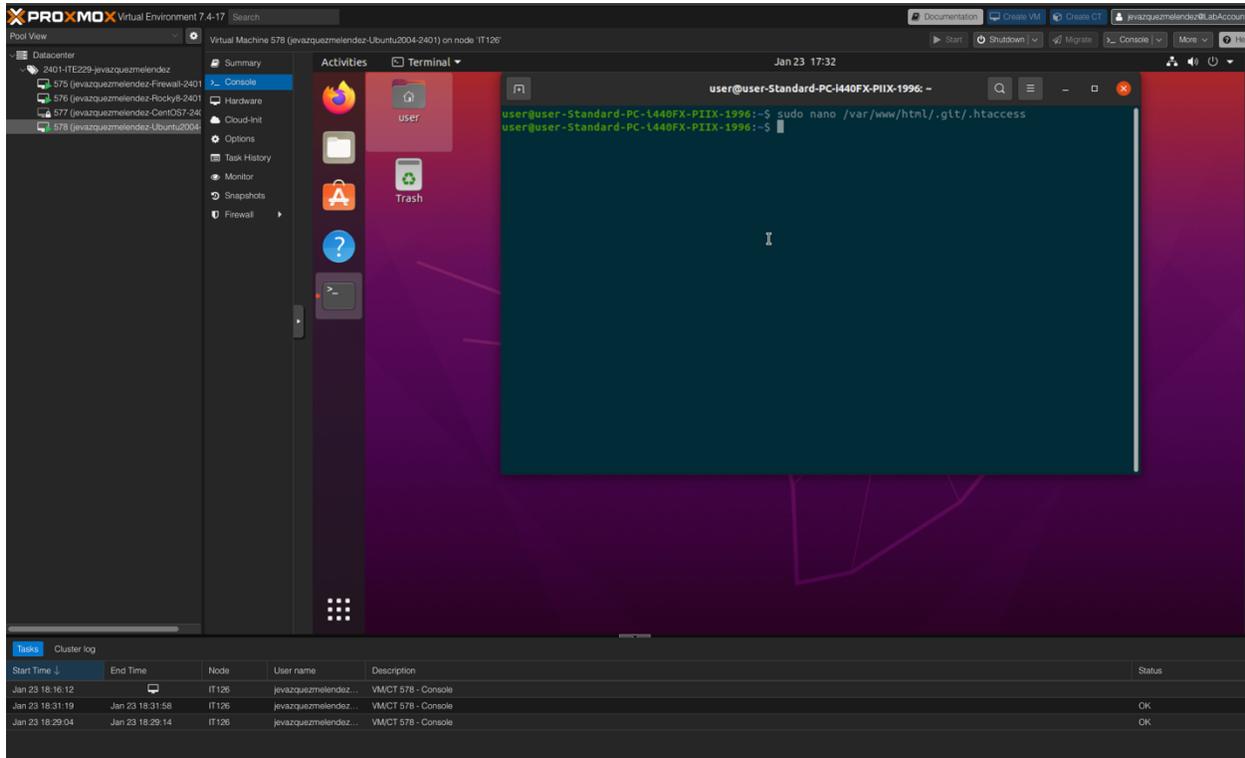
Start Time	End Time	Node	User name	Description	Status
Jan 23 18:24:45		IT126	jevaizquizmelendez...	VMCT 578 - Console	
Jan 23 18:16:12		IT126	jevaizquizmelendez...	VMCT 578 - Console	

Create a .htaccess File in the /var/www/html/.git/ Directory

```
$ sudo nano /var/www/html/.git/.htaccess
```

The command `sudo nano /var/www/html/.git/.htaccess`

This creates using the Nano text editor the file ".htaccess" located in the "/var/www/html/.git/" directory. This file is used to configure access restrictions for the Apache web server.

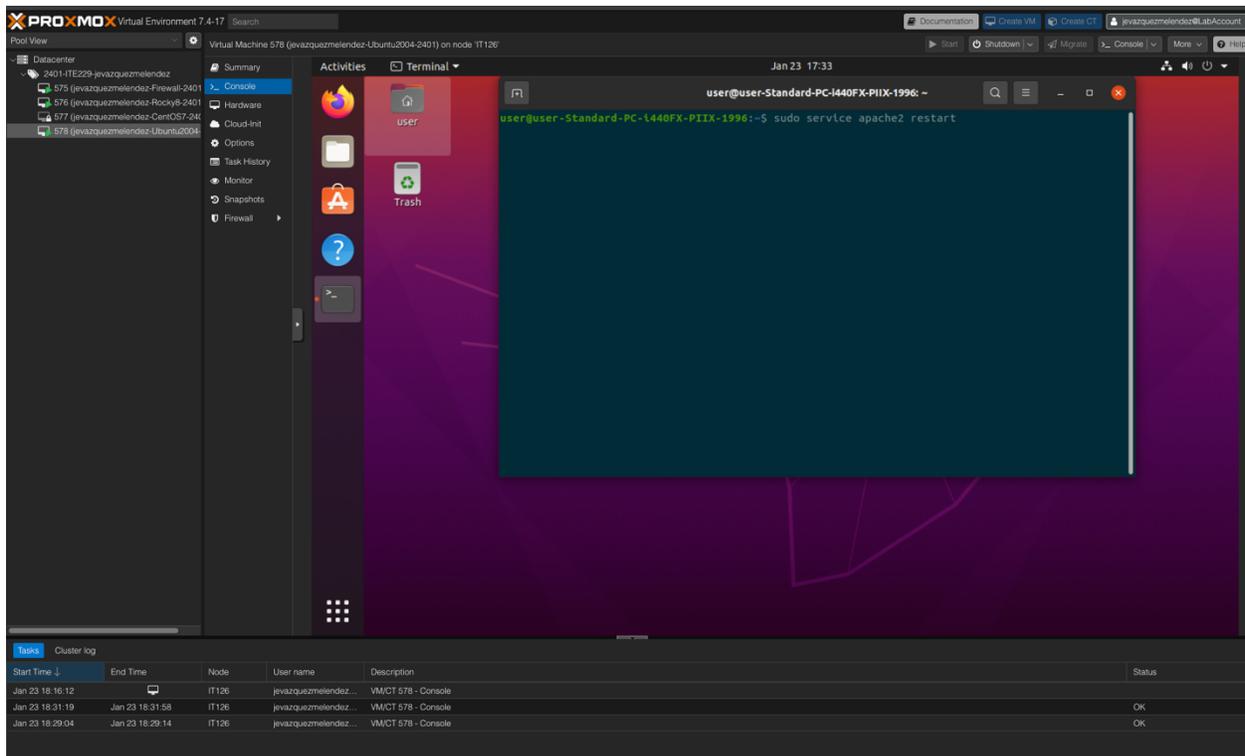


Restart the Apache Service

```
$ sudo service apache2 restart
```

Using this command `sudo service apache2 restart`. Restart the Apache web server.

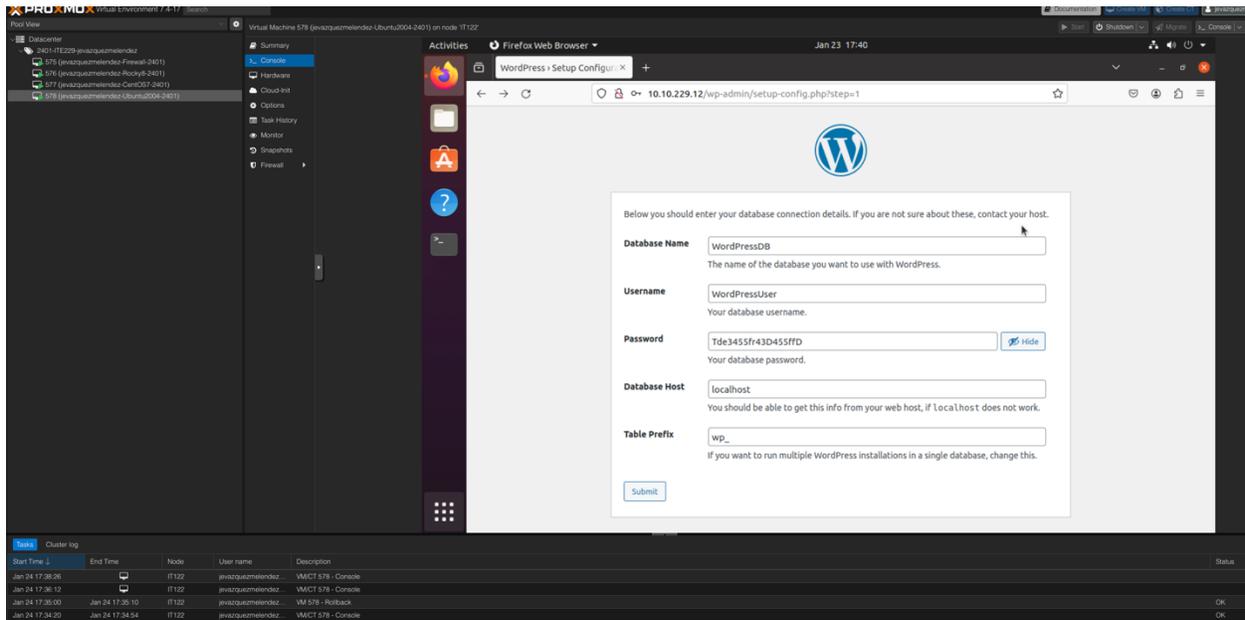
This is necessary when changes are made to Apache's configuration files, such as the `.htaccess` file mentioned above, to ensure that the changes take effect.



The screenshot displays the Proxmox Virtual Environment (VE) interface. The main window shows a terminal session for a virtual machine named 'user@user-Standard-PC-1440FX-PIIX-1996'. The terminal prompt is '\$', and the command 'sudo service apache2 restart' has been entered. The background of the terminal window shows a desktop environment with a purple and blue gradient, a 'user' window, and a 'Trash' icon. The Proxmox interface includes a sidebar with navigation options like 'Datacenter', 'Summary', 'Hardware', 'Cloud-Init', 'Options', 'Task History', 'Monitor', 'Snapshots', and 'Firewall'. At the bottom, there is a 'Tasks' table with the following data:

Start Time	End Time	Node	User name	Description	Status
Jan 23 18:16:12		IT126	jevizquezmelendez...	VMCT 578 - Console	
Jan 23 18:31:19	Jan 23 18:31:58	IT126	jevizquezmelendez...	VMCT 578 - Console	OK
Jan 23 18:29:04	Jan 23 18:29:14	IT126	jevizquezmelendez...	VMCT 578 - Console	OK

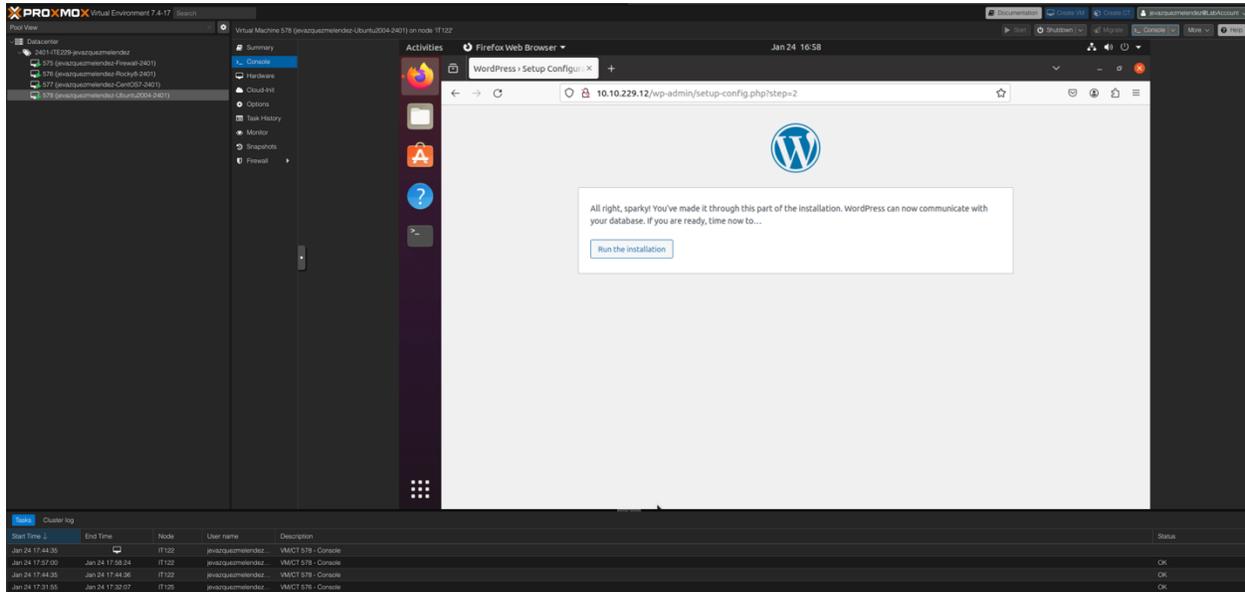
Then we must type the Data Base information we provided.
This will allow us to go ahead and proceed with the installation process.



- **Database Name:** WordPressDB
- **Username:** WordPressUser
- **Password:** Tde3455Fr43D455ffD
- **Database Host:** localhost
- **Table Prefix:** wp_

Run Installation

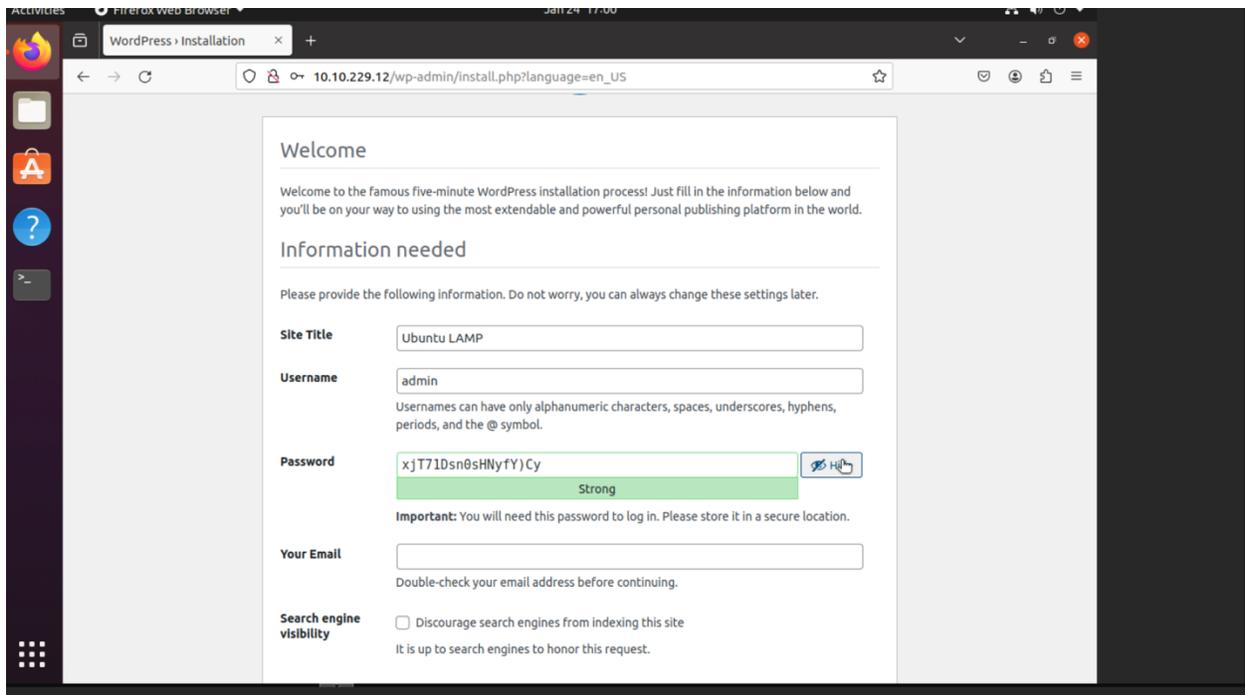
This is the prompt to RUN THE INSTALLATION of Word Press once the data base information is mapped correctly to the LAMP Stack.



Create an Admin WordPress User

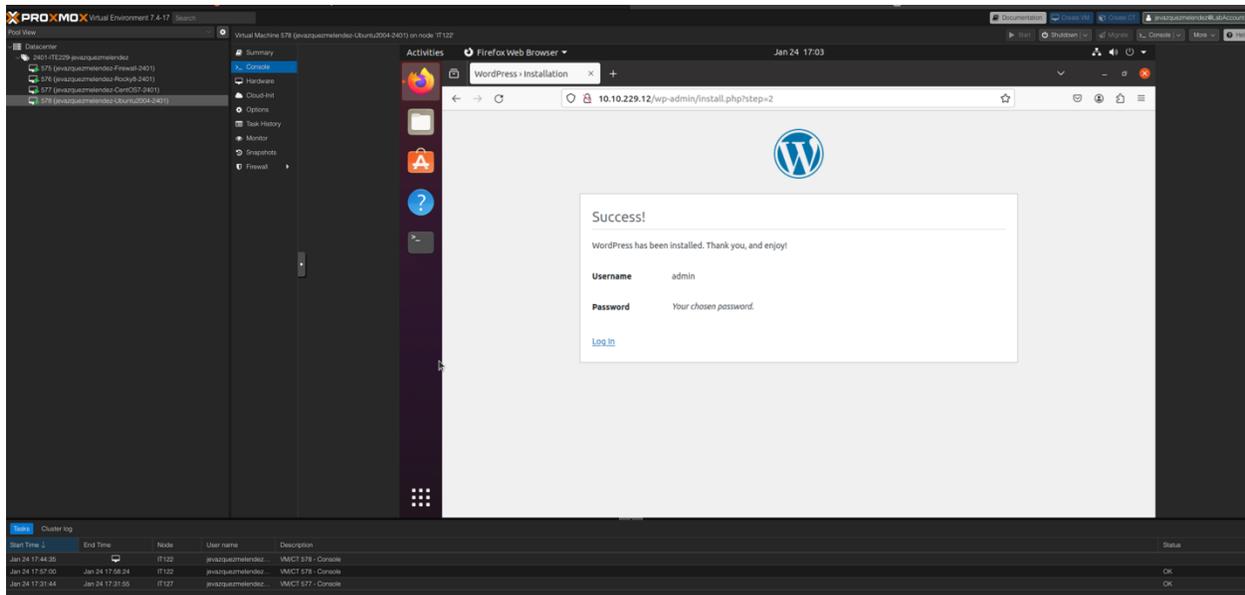
We need to create a WordPress User account.

To Create an account, Enter the information prompted.



Success!

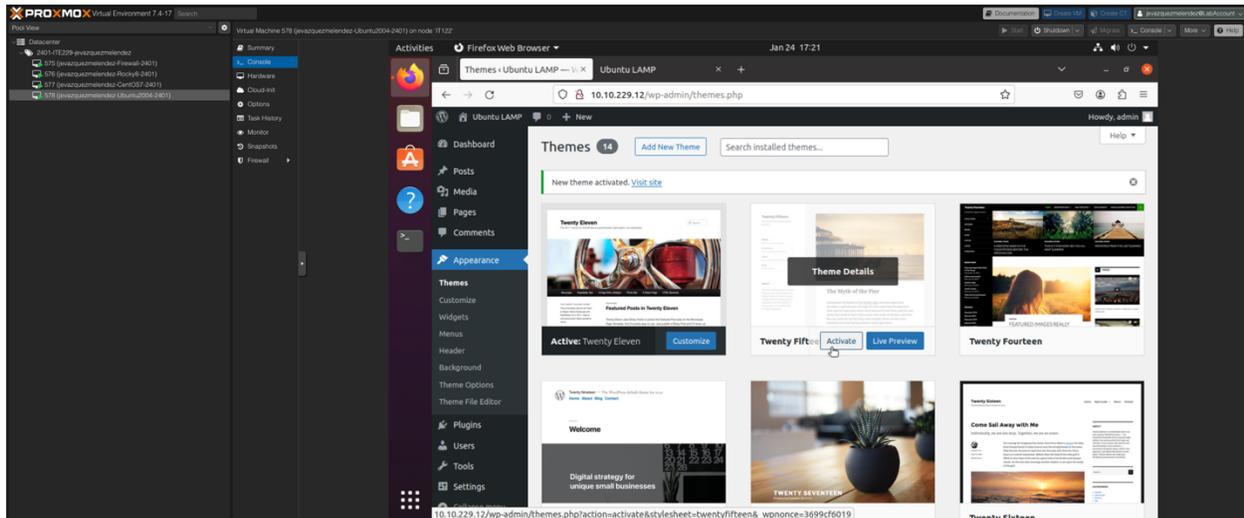
The credentials are successfully created, and you can go ahead and press Log in.



WordPress Site Selections

After successfully Logged in the WordPress Account we just created we can see the WordPress Dashboard.

To change the appearance, scroll to the Paint Brush that shows “Appearance” and a “Themes” tab should appear press it to show the default website pages you can select from.



Site Title: Ubuntu LAMP

Username: admin

Password: *xjT71Dsn0sHNyfy)Cy*

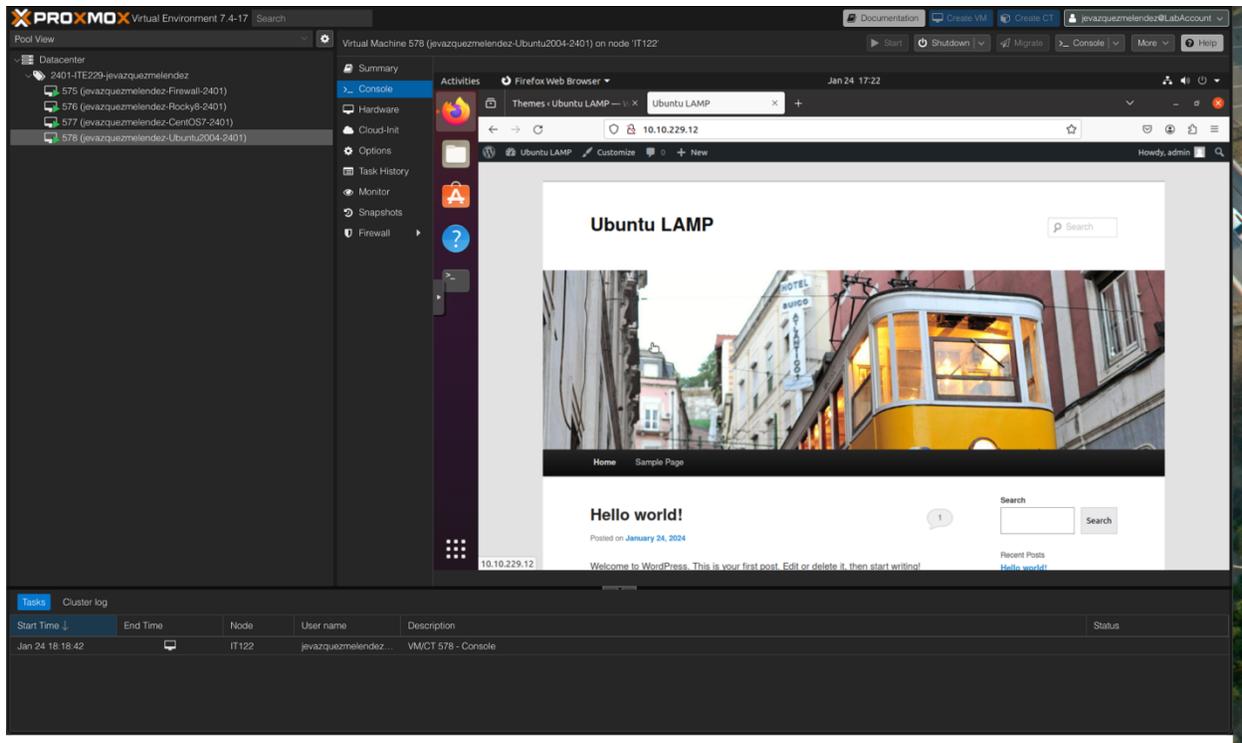
Your email: root@localhost.local

Search Engine Visibility: leave unchecked

Test WordPress

To Test the Page, open a new Firefox Web Browser window, and type <http://10.10.229.12>

The main page with the tittle “Ubuntu LAMP” should appear as all the setting are properly configured.



The screenshot displays the Proxmox Virtual Environment (VE) interface. The main window shows a virtual machine console for 'Virtual Machine 578 (javazquezmelendez-Ubuntu2004-2401) on node: IT122'. The console is running a Firefox web browser at the IP address 10.10.229.12. The browser displays the 'Ubuntu LAMP' WordPress homepage, which features a header with the title 'Ubuntu LAMP', a search bar, and a main content area with a post titled 'Hello world!' dated 'January 24, 2024'. The Proxmox interface includes a sidebar with navigation options like 'Summary', 'Console', 'Hardware', and 'Options'. At the bottom, there is a 'Tasks' table showing a recent task.

Start Time	End Time	Node	User name	Description	Status
Jan 24 18:18:42		IT122	javazquezmelendez...	VM/CT 578 - Console	

IMPORTANT: To earn full credit, original formatting of this document must remain in place. In addition, ALL screenshots must include a “full view”, including your ProxMox title bar with your username showing.

Milestone #3: WordPress Security & Configurations

WORDPRESS SECURITY SUMMARY

Exemplary: HALF PAGE WordPress security summary FULLY DISCUSSING what you secured on your blog site. **MINIMUM: 300 words required—not including references. Include at least two (2) research references.**

DEFENSE-IN-DEPTH

Exemplary: HALF PAGE FULLY discussing how you applied defense-in-depth to this exercise.
MINIMUM: 300 words required—not including references. Include at least two (2) research references.

FILE PERMISSIONS

Before and After Screenshots AND written instructions required; Include at least two (2) sentences per screenshot.

- a. **Vulnerability** (show screenshot and explain why this is a vulnerability)
- b. **Configuration** (show screenshot and explain the steps you took to config changes)
- c. **Validation** (show screenshot and explain the step you took to validate/confirm)

SECURING WP-CONFIG.PHP

Before and After Screenshots AND written instructions required; Include at least two (2) sentences per screenshot.

- a. **Vulnerability** (show screenshot and explain why this is a vulnerability)
- b. **Configuration** (show screenshot and explain the steps you took to config changes)
- c. **Validation** (show screenshot and explain the step you took to validate/confirm)

FIREWALL (Shield)

Before and After Screenshots AND written instructions required; Include at least two (2) sentences per screenshot.

- a. **Vulnerability** (show screenshot and explain why this is a vulnerability)
- b. **Configuration** (show screenshot and explain the steps you took to config changes)
- c. **Validation** (show screenshot and explain the step you took to validate/confirm)

CONCLUSION

Exemplary: HALF PAGE conclusion FULLY DISCUSSING how you applied security methodology and how you might apply it to other systems. MINIMUM: **300 words required—not including references.** **Include at least two (2) research references.**

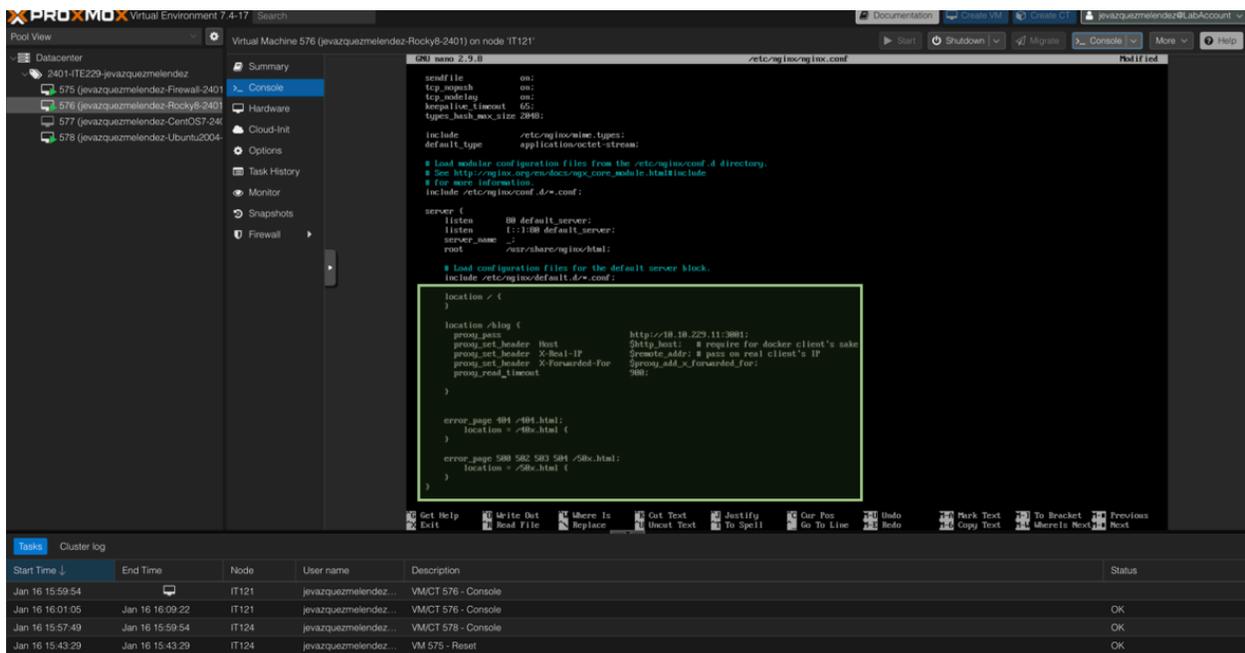
IMPORTANT: To earn full credit, original formatting of this document must remain in place. In addition, ALL screenshots must include a “full view”, including your ProxMox title bar with your username showing.

Appendix A - (Milestone #1)

NginX Config File (Include screenshot & explain)

You must follow a few simple steps to edit thx configuration file for a web server like Ghost. First, you need to locate the configuration file on your server. This file is typically located in the /etc/nginx/nginx.conf directory. Once you have located the file, you can open it using a text editor Nano.

Next, you must find the relevant section of the configuration file that you want to modify. This could include server settings, SSL configurations, or proxy settings. Once you have located the relevant section, you can make the necessary changes to the file.



```
Proxmox Virtual Environment 7.4-17 | Search | Documentation | Create VM | Create CT | jvazquezmelendez@LabAccount |
Pool View | Virtual Machine 576 (jvazquezmelendez-Rocky8-2401) on node IT121 | Start | Shutdown | Migrate | Console | More | Help
Datacenter | 2401-ITE229-jvazquezmelendez | 575 (jvazquezmelendez-Firewall-2401) | 576 (jvazquezmelendez-Rocky8-2401) | 577 (jvazquezmelendez-CentOS7-24) | 578 (jvazquezmelendez-Ubuntu2004)
Summary | Console | Hardware | Cloud-Init | Options | Task History | Monitor | Snapshots | Firewall
GNU nano 2.9.3 | /etc/nginx/nginx.conf |
# nginx.conf
#
# Example configuration for an http server
#
# Load modular configuration files from the /etc/nginx/conf.d directory.
# See http://nginx.org/en/docs/http_module.html#http_module for more information.
# For more information
include /etc/nginx/conf.d/*.conf;

server {
    listen 80 default_server;
    listen [::]:80 default_server;
    server_name _;
    root /usr/share/nginx/html;

    # Load configuration files for the default server block.
    include /etc/nginx/default.d/*.conf;

    location / {
    }

    location /blog {
        proxy_pass http://10.10.229.11:8081;
        proxy_set_header Host $http_host; # require for Docker client's sake
        proxy_set_header X-Real-IP $remote_addr; # pass on real client's IP
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_read_timeout 90s;
    }

    error_page 499 /499.html;
    location = /49x.html {
    }

    error_page 500 502 503 504 /50x.html;
    location = /50x.html {
    }
}
Get Help | Exit | Write Out | Read File | Undo | Replace | Cut Text | Uncut Text | Justify | To Spell | Our Pos | Go To Line | Undo | Redo | Mark Text | Copy Text | To Bracket | Where's Next | Previous | Next
Tasks | Cluster log
Start Time | End Time | Node | User name | Description | Status
Jan 16 15:59:54 | | IT121 | jvazquezmelendez... | VMCT 576 - Console |
Jan 16 16:01:05 | Jan 16 16:09:22 | IT121 | jvazquezmelendez... | VMCT 576 - Console | OK
Jan 16 15:57:49 | Jan 16 15:59:54 | IT124 | jvazquezmelendez... | VMCT 578 - Console | OK
Jan 16 15:43:29 | Jan 16 15:43:29 | IT124 | jvazquezmelendez... | VM 575 - Reset | OK
```

Appendix B - (Milestone #1)

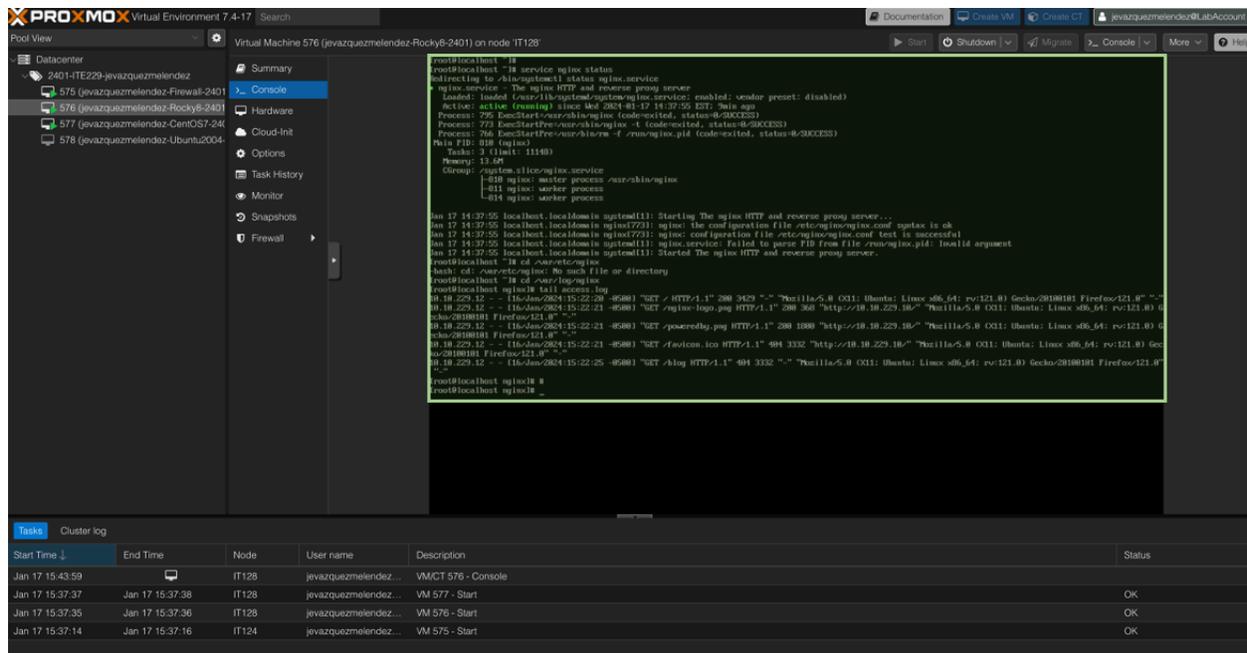
NginX Access Log File

```
# cd /var/log/nginx
```

This command "cd /var/log/nginx" is used to change the current directory to "/var/log/nginx". "cd" stands for "change directory" and "/var/log/nginx" is the path of the directory we want to change to.

```
# tail access.log
```

This command is used to display the last few lines of the "access.log" file. "tail" is a command that displays the last few lines of a file, and "access.log" is the name of the file we want to display. By default, "tail" displays the last 10 lines of a file.



```
root@localhost:~# service nginx status
nginx.service - The nginx HTTP and reverse proxy server
Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; vendor preset: disabled)
Active: active (running) since Wed 2024-01-17 14:37:55 EST; 5min ago
Process: 795 ExecStart=/usr/sbin/nginx (code=exited, status=0/SUCCESS)
Process: 773 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCCESS)
Process: 766 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, status=0/SUCCESS)
Main PID: 818 (nginx)
Tasks: 3 (limit: 11169)
Memory: 13.6M
Group: system.slice/nginx.service
├─818 nginx: master process /usr/sbin/nginx
└─814 nginx: worker process

Jan 17 14:37:55 localhost.localdomain systemd[1]: Starting The nginx HTTP and reverse proxy server...
Jan 17 14:37:55 localhost.localdomain nginx[773]: nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
Jan 17 14:37:55 localhost.localdomain nginx[773]: nginx: configuration file /etc/nginx/nginx.conf test is successful
Jan 17 14:37:55 localhost.localdomain systemd[1]: nginx.service: failed to parse PID from file /run/nginx.pid: Invalid argument
Jan 17 14:37:55 localhost.localdomain systemd[1]: Started The nginx HTTP and reverse proxy server.
root@localhost:~# cd /var/log/nginx
bash: cd: /var/log/nginx: No such file or directory
root@localhost:~# cd /var/log/nginx
root@localhost:~# tail -n 10 /var/log/nginx/access.log
18.18.229.12 - [18-Jan-2024:15:22:29 -0900] "GET / HTTP/1.1" 200 3429 "-" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:121.0) Gecko/20100818 Firefox/121.0" "-"
18.18.229.12 - [18-Jan-2024:15:22:21 -0900] "GET /nginx-logo.png HTTP/1.1" 200 368 "http://18.18.229.18/" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:121.0) Gecko/20100818 Firefox/121.0" "-"
18.18.229.12 - [18-Jan-2024:15:22:21 -0900] "GET /poweredu.png HTTP/1.1" 200 1088 "http://18.18.229.18/" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:121.0) Gecko/20100818 Firefox/121.0" "-"
18.18.229.12 - [18-Jan-2024:15:22:21 -0900] "GET /favicon.ico HTTP/1.1" 404 3332 "http://18.18.229.18/" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:121.0) Gecko/20100818 Firefox/121.0" "-"
18.18.229.12 - [18-Jan-2024:15:22:25 -0900] "GET /blog HTTP/1.1" 404 3332 "-" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:121.0) Gecko/20100818 Firefox/121.0" "-"
root@localhost:~#
root@localhost:~#
```

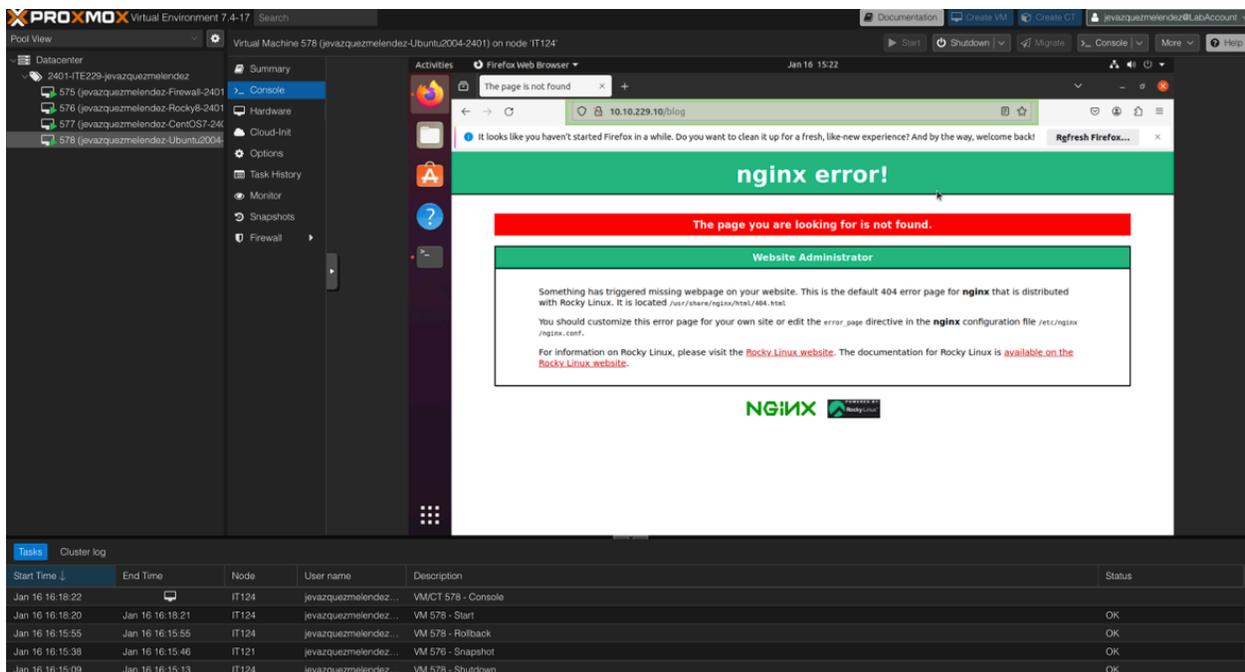
Start Time	End Time	Node	User name	Description	Status
Jan 17 15:43:59		IT128	jevaquezmelendez...	VM/CT 576 - Console	
Jan 17 15:37:37	Jan 17 15:37:38	IT128	jevaquezmelendez...	VM 577 - Start	OK
Jan 17 15:37:35	Jan 17 15:37:36	IT128	jevaquezmelendez...	VM 576 - Start	OK
Jan 17 15:37:14	Jan 17 15:37:16	IT124	jevaquezmelendez...	VM 575 - Start	OK

NginX Error Log File

Page Not Found error on a website means that the web server cannot locate the requested resource or file. This error is commonly known as a 404 error, and it occurs when the web server fails to find the requested page or resource on the server.

As a website administrator, you can do a few things to troubleshoot a 404 error. First, you should check the URL to ensure it is correct and point to the correct file or resource. You should also check the server logs to see if any error messages can give you a clue as to what went wrong.

If the issue is due to a deleted or moved file, you should restore the file or update the URL to point to the correct location. If the problem is due to an incorrect server configuration, you should review the configuration files and make any necessary changes.



The screenshot shows a Proxmox VE interface with a Firefox browser window open. The browser displays a 404 error page with the following content:

```
nginx error!  
The page you are looking for is not found.  
Website Administrator  
Something has triggered missing webpage on your website. This is the default 404 error page for nginx that is distributed with Rocky Linux. It is located /usr/share/nginx/html/404.html  
You should customize this error page for your own site or edit the error_page directive in the nginx configuration file /etc/nginx/nginx.conf.  
For information on Rocky Linux, please visit the Rocky Linux website. The documentation for Rocky Linux is available on the Rocky Linux website.  
NGINX
```

The browser address bar shows the URL `10.10.229.10/blog`. The Proxmox interface includes a sidebar with navigation options like Summary, Console, Hardware, and Cloud-Init, and a task log at the bottom.

Start Time	End Time	Node	User name	Description	Status
Jan 16 16:18:22		IT124	jevaquzmelendez...	VM/CT 578 - Console	
Jan 16 16:18:20	Jan 16 16:18:21	IT124	jevaquzmelendez...	VM 578 - Start	OK
Jan 16 16:15:55	Jan 16 16:15:55	IT124	jevaquzmelendez...	VM 578 - Rollback	OK
Jan 16 16:15:38	Jan 16 16:15:46	IT121	jevaquzmelendez...	VM 576 - Snapshot	OK
Jan 16 16:15:09	Jan 16 16:15:13	IT124	jevaquzmelendez...	VM 578 - Shutdown	OK